

Mika Luoma

3D-älykameran soveltaminen miehittämättömän maa-ajoneuvon autonomiseen ohjaukseen ja lastinkäsittelyn automatisointi

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Konetekniikka

Insinöörityö

25.4.2018

Alkusanat

Työn suorittajalla on neljännesvuosisadan työkokemus raskaiden ajoneuvojen ja työkoneiden hydraulikkapiirien, ohjausjärjestelmien sekä ajoneuvosähköjen asennuksista ja huollosta. Mielenkiintoisempaa insinöörityötä ei olisi tässä voinut kohdalleni sattua. Kaikki aiempi työelämässä kertynyt kokemus on hyödynnettävissä, ja usean vuoden uudelleen koulutuksen tuomat uudet taidot punnitaan.

Haluan kiittää Rakkatec Oy:n henkilökuntaa, etenkin systeemiasiantuntija Jere Knuuttilaa logiikkojen ohjelmointityön tuesta ja ohjauksesta, IFM:n tuotespesialisteja Dan Eklundia ja Aarno Kauppia 3D/2D-älykameraan, näyttömoduuliin ja antureihin liittyvästä teknisestä tuesta, Technionin asiantuntijaa Tommi Sairoa logiikan teknisestä tuesta ja Codesys 2.3:n ohjelmointiympäristön neuvonnasta sekä Metropolian lehtoria Antti Liljaniemeä vaativista kursseista ja ohjauksesta. Kaikille tasapuolisesti kiitos korvaamattomasta avusta ja asiantuntemuksesta.

Helsingissä 25.4.2018

Mika Luoma

Tekijä Otsikko Sivumäärä Aika	Mika Luoma 3D-älykameran soveltaminen miehittämättömän maa-ajoneuvon autonomiseen ohjaukseen ja lastinkäsittelyn automatisointi 62 sivua + 5 liitettä 25.4.2018
Tutkinto	Insinööri (AMK)
Tutkinto-ohjelma	Konetekniikka
Ammatillinen pääaine	Koneautomaatio
Ohjaajat	System Specialist Jere Knuuttila Lehtori Antti Liljaniemi
<p>Rakkatec Oy kehittää, suunnittelee ja valmistaa miehittämättömiä maastoajoneuvoja (UGV), joita ohjataan radioyhteyksillä lähietäisyyksiltä. Ajoneuvon alusta on herättänyt mielenkiintoa kriisienhallintatehtävissä, mm. katastrofialueille avun toimittamiseksi, metsäpalojen sammutustöissä ja muissa tehtävissä, joihin ihmisen lähettäminen koneen mukana olisi hengenvaarallista tai mahdotonta. Yksi tärkeimmistä ominaisuuksista etäohjauksen lisäksi on laitteen kyky paikantaa, noutaa ja toimittaa erilaisia operaatioihin tarvittavia moduuleita ilman operaattorin näköyhteyttä. Valmista järjestelmää kyseiseen tarpeeseen ei ole löytynyt markkinoilta.</p> <p>Tässä insinööriyössä asennettiin ajoneuvon alustalle tarvittava laitteisto ja suoritettiin ohjelmointityö, sisältäen lähietäisyydellä olevan moduulin havaitseminen 3D-älykameralla, ajoneuvon siirto autonomisesti nostopaikalle, moduulin autonominen nosto tai lasku asematietoon perustuen tai operaattorin pyynnöstä ilman näköyhteyttä.</p> <p>Toteutuksessa hyödynnettiin IFM O3M 251-älykameraa ja IFM CR1083-näyttömoduulia, kahta IFM O3M-älykameralle kehitettyä apuohjelmaa ja omaa ohjelmaa, joka prosessoi tietoa ja välittää tarvittavat ohjaustiedot koukkulaitteen ja alustan logiikoille. Lavan käsittelyyn käytettiin CPS-koukkulavanostinta, joka varusteltiin tarvittavilla antureilla ja automatisoitiin Technionin TEC132 logiikalla Codesys 2.3-versiolla. Alustana käytettiin Rakka 3000 nivelohjattua hydrostaattista alustaa, jolle asennettiin tarvittava runkonivelen asennon mittaus, ja koukkulaitteiston tarvitseman hydraulikkapaineen ulosotto kytkettiin alustan proportionaaliventtiiliryhmään.</p>	
Avainsanat	IFM O3M 251, IFM CR 1083, 3D-älykamera, TEC 132, autonominen ajoneuvo, autonominen lastinkäsittely

Author Title	Mika Luoma 3D Smart Sensor Guided Unmanned Ground Vehicle and the Automation of Cargo Handling
Number of Pages Date	62 pages + 5 appendices 25 April 2018
Degree	Bachelor of Engineering
Degree Programme	Mechanical Engineering
Professional Major	Machine Automation
Instructors	Jere Knuuttila, System Specialist Antti Liljaniemi, Senior Lecturer
<p>Rakkatec OY develops, designs and manufactures UGV (Unmanned Ground Vehicles) which are controlled at a close range via radio controllers of the customers choice. The vehicle has aroused interest especially in the fields of crisis control, e.g. the delivery of supplies to catastrophe areas, forest fire extinguishing and many other missions where sending people with a machine would be extremely dangerous or possibly fatal. A very important feature required for an efficient operation, is the ability to recognize, retrieve and deliver different types of modules essential for a successful mission, without a visual contact and operator interaction. At the moment, however, there are no setups with the needed abilities available on the market.</p> <p>The required hardware and software were installed during this project that enable a close range detection of a module, transportation of the vehicle to the destination, lifting and dropping of module.</p> <p>In the implementation, an IFM O3M 251 Smart Sensor and IFM CR1083 Graphic Display Module are utilized along with two IFM demo programs developed for a reflector recognition and a custom, self developed program for the guidance of the UGV. The module is programmed to process data received from the camera system and transmit the needed guidance messages to the logics of the CPS hooklift and UGV. For module handling, a CPS hooklifting device is installed with sensors required to meet the automation demands and controlled by a Technionin TEC132 logic controller and programmed in the Codesys 2.3 programming environment. The UGV is the articulate-steered and hydrostatic driven Rakka 3000. The articulated steering is supplied with a position sensor. Hydraulic pressure for the hooklift is supplied using a proportional hydraulic valve from the vehicle.</p>	
Keywords	IFM O3M 251, IFM CR 1083, 3D Smart Sensor, TEC 132, autonomous vehicle, autonomous cargo handling

Sisällys

Lyhenteet

1	Johdanto	1
1.1	Työn kuvaus	1
1.2	Rajaukset	2
2	Konenäkö, kuvankäsittelyn teoriaa, 3D-kuvaus	3
2.1	Konenäkö	5
2.2	Kuvankäsittely (Image Processing)	8
2.2.1	Kynnysvertailu (Thresholding)	10
2.2.2	Reunantunnistus (Edge Detection)	11
2.2.3	Blob-tunnistus (Blob Detection)	11
2.2.4	Hahmontunnistus (Pattern Recognition)	11
2.3	3D-kuvaus ja Time of Flight kamera	12
2.3.1	3D-kuvauksen perusteita	12
2.3.2	3D älyanturin käyttö ja mahdolliset sovellukset	16
3	Projektissa käytetyt laitteet ja komponentit	19
4	IFM O3M 251-kameran ohjelmointi, apuohjelmat ja ohjelmointilaitteet	22
4.1	Visual Assistant	22
4.2	3D/2D-anturin asennus	23
4.3	Kameran konfigurointi ja parametrisointi	25
5	Kallistusanturin konfigurointi	27
6	Logiikkojen kytkennät ja ohjelmointi	29
6.1	Technion TEC132-kytkennät ja konfigurointi	29
6.2	Ohjelmointi	31
6.3	Alustan paikoitus nostoasemalle	32
6.4	Nostosekvenssi	35
6.5	Laskusekvenssi	36
7	IFM CR1083-näyttömoduulin kytkennät ja konfigurointi	38

7.1	Ohjelmointi ja apuohjelmat	39
7.2	Järjestelmän eri CAN-protokollat	42
7.2.1	CAN Open	42
7.2.2	Raw CAN ja J1939	42
8	Ohjelmien arkkitehtuuri ja suoritus	43
8.1	CR1083 PLC ja O3M 251-anturi	43
8.2	Technion TEC 132	45
9	Hydrauliikka	50
9.1	CPS-Koukkulaite (Hook Lift)	50
9.2	Rakka 3000 UGV	52
10	Sähkökytkennät	53
10.1	CPS-koukkulaite	53
10.2	Rakka 3000 UGV alusta	53
11	Lavan varustelu ja mekaaniset muutokset	54
12	Tulosten arviointi ja jatkokehitysehdotukset	56
12.1	Koukkulaitteen automatisointi	56
12.2	Moduulin paikannus ja UGV:n asemointi	56
12.3	Jatkokehitysehdotukset	58
13	Yhteenveto	59
	Lähteet	61
	Liitteet	
	Liite 1. Ohjelmointityössä käytettyjä linkkejä	
	Liite 2. IFM CR 1083 Module retrieve drive program	
	Liite 3. Technion TEC 132 Hooklift and UGV steering program	
	Liite 4. Hydrauliikkakaavio	
	Liite 5. Sähkökytkentöjen kytkentälista	

Lyhenteet

AIA	<i>Automated Imaging Association</i> , maailmanlaajuinen konenäköön ja kuvankäsittelyyn erikoistunut ammattilaisten ryhmittymä
Blob	<i>Binary Large Object</i> , kuvankäsittelyssä käytetty termi alueesta, jossa on ryhmittymä yhdenmukaista binääristä dataa
CAN	<i>Controller Area Network</i> , automaatioväylä, jota käytetään ajoneuvoissa ja teollisuudessa välittämään tietoa
DI	<i>Distance Image</i> , etäkuvan havainnointi
FB	<i>Function Block</i> , funktiolohko
FBD	<i>Function Block Diagram</i> , funktiolohko-ohjelma
FOV	<i>Field of View</i> , näkökenttä
IE	<i>Industrial Ethernet</i> , teollinen Ethernet
I/O	<i>Input/Output</i> , sisääntulo ja ulosmeno
LG	<i>Line Guidance</i> , viivan tunnistus ja seuranta
OCR	<i>Optical Character Recognition</i> , tekstintunnistus
OCV	<i>Optical Character Verification</i> , tekstintunnistuksen vahvistus
OD	<i>Object Detection</i> , objektin tunnistus
PID	<i>Proportional-integral-derivative-regulator</i> ,proportionaali-integroiva-derivoiva-säädin

PLC	<i>Programmable Logic Controller</i> , ohjelmoitava logiikkaohjain
PMD	<i>Photonic Mixer Device</i> , teknologia, jolla mitataan etäisyyttä reaaliajassa
ST	<i>Structured Text</i> , strukturoitu teksti Codesys ohjelmointi ympäristössä
ToF	<i>Time of Flight</i> , etäisyyden mittausmenetelmä, joka perustuu valon takaisin heijastuvaan aikaan, objektin ja vastaanottimen välillä
UGV	<i>Unmanned Ground Vehicle</i> , miehittämätön maa-ajoneuvo

1 Johdanto

1.1 Työn kuvaus

Tämän insinöörityön tarkoituksena on suunnitella, varustella ja ohjelmoida Rakka 3000-miehittämättömän maa-ajoneuvon lastinkäsittelylle autonominen järjestelmä, joka toimii lähietäisyyksillä (2–10 m). Järjestelmään kuuluu IFM O3M-sarjan Smart Sensor paikantamaan moduuleita ja välittämään asematietoja. Teoriaosuudessa käydään läpi konenäköä, sen kuvankäsittelyä suppeasti ja projektissa käytetyn laitteiston 3D/2D-älykameran 3D-mittaukseen perustuvaa tekniikkaa ja toimintaperiaatetta.

Lastinkäsittelyn lähtökohtana pidetään tilannetta, jossa ajoneuvo on sijoittunut operaattorin toimesta tai tulevaisuudessa etäohjattuna alueelle, jossa tiedetään noudettavan lavan, kontin tai moduulin sijaitsevan. Kohteen on sijaittava käytössä olevan järjestelmän havaintoalueella. Vaihteleva valoisuus tai pimeys ei saa olla esteenä, ja esimerkiksi sumua, pölyä ja savua voi esiintyä kohtuullisesti. Maastona tulee olla tasainen piha tai muu vastaava alue.

Moduulinkäsittelyn osuudessa, automatisointityössä varustellaan alustan päälle asennettu CPS-koukkulaite tarvittavalla anturoinnilla ja Technionin TEC132-logiikkaohjaimella. Moduulinkäsittelylle luodaan automaattiset moduulin päälle nostoon ja pois laskemiseen tarvittavat sekvenssit, joissa on huomioituna toimintoja, jotka normaaleissa tilanteissa ovat näköyhteyttä pitävän operaattorin hallinnassa, kuten koukun moduuliin kiinnittyminen, moduulin onnistunut asettuminen alustan ohjausrullille ja lukituksien tilat. Ohjelmointiympäristönä toimii Codesys 2.3.

Alustan asemoimiseen implementoidaan IFM O3M 251 Smart Sensor, joka paikantaa noudettavat lavat tai moduulit ja välittää IFM CR1083-näyttömoduulin avulla asematietoa ja ohjaustietoa Technion TEC132-logiikalle, jossa suoritetaan sekä moduulinkäsittelyn ohjaus että prosessoidaan ohjauskomennot ajosuunnasta ja nopeudesta alustan Exertus-logiikalle.

Työssä hyödynnetään kahta IFM:n kehittämää heijastimia paikantavaa apuohjelmaa: AGV Follow Sample Project Version 1.1, jota käytetään autonomisilla satamakonttien siirtoalustoilla, ja ifm_O3M_OD_V041906 Object_List, joka havaitsee ja tulostaa

visuaaliseen taulukkoon ensin heijastavat esineet ja yhteensä kahdeksan lähimpänä olevaa objektia, laskien näiden tarkat koordinaatit. Menetelmänä moduulin paikannukseen käytetään tietyillä mitoilla toisistaan poikkeavien ja samalla korkeudella olevien heijastimien seuraamista kolmiomittauksen perusteella.

1.2 Rajaukset

Työ rajataan siten, että havaintoalue on kohteesta maksimissaan 10 m:n etäisyydellä ja sallittu poikkeama kohteen etupään keskeltä mitattuna, etäisyys/poikkeama, on sellainen, että asemointi ajoneuvon alustalla onnistuu yhdellä etenemisellä, ilman edestakaista korjausliikettä.

Alustan Exertus-logiikkaan tarvittavista ohjelman muutoksista vastaa Rakkatec OY. Lastin käsittelyn ohjelmoinnissa ei huomioida keskeytystilanteita, hätätoimintoja tai noston/laskun mahdollista epäonnistumista.

Laitteen turvallisuustoimenpiteenä, liitetään alustan hätäpysäytyspiirin signaali CAN-viestinä, joka huomioidaan ohjelmallisesti omassa koodissa. Laitteen autonomiset liikkeet ovat käyttäjän hallinnassa Hetronic radio-ohjaimen välityksellä, niin että autonomia pidetään jonkin katkaisijan tms. välityksellä fyysisesti pakotettuna, esimerkiksi jonkin katkaisijan päällä pitämisellä, siten että se vapautuu, kun käyttäjän kontakti puuttuu.

2 Konenäkö, kuvankäsittelyn teoriaa, 3D-kuvaus

Konenäkö, robottinäkö, älykamera sekä skannaavia ja kuvia tulkitsevia järjestelmiä on vyörynyt markkinoille vuosituhannen vaihteen jälkeen. Automated Imaging Associationin (AIA) mukaan alalla toimii yli neljäsataa eri laitetoimittajaa [1, s. 5].

Eri mittausperiaatteilla ja teknisillä ratkaisuilla valmistetuilla komponenteilla ja järjestelmillä on kullakin omat etunsa ja rajoituksensa. Tuotantolinjan osana asemointi- tai mittaustehtäviin suunnitellulta laitteistolta vaaditaan eri ominaisuuksia, verrattuna valvontatehtäviin tarkoitetun laitteiston. Älyanturin toiminta ulkona sääolosuhteiden armoilla, vaihtelevassa valossa ja näkyvyydessä, haastaa järjestelmän suunnittelua eri tasolla, kuin sisätiloissa toimivan autonomisesti suoriutuvan trukin navigointi 3D-anturilla.

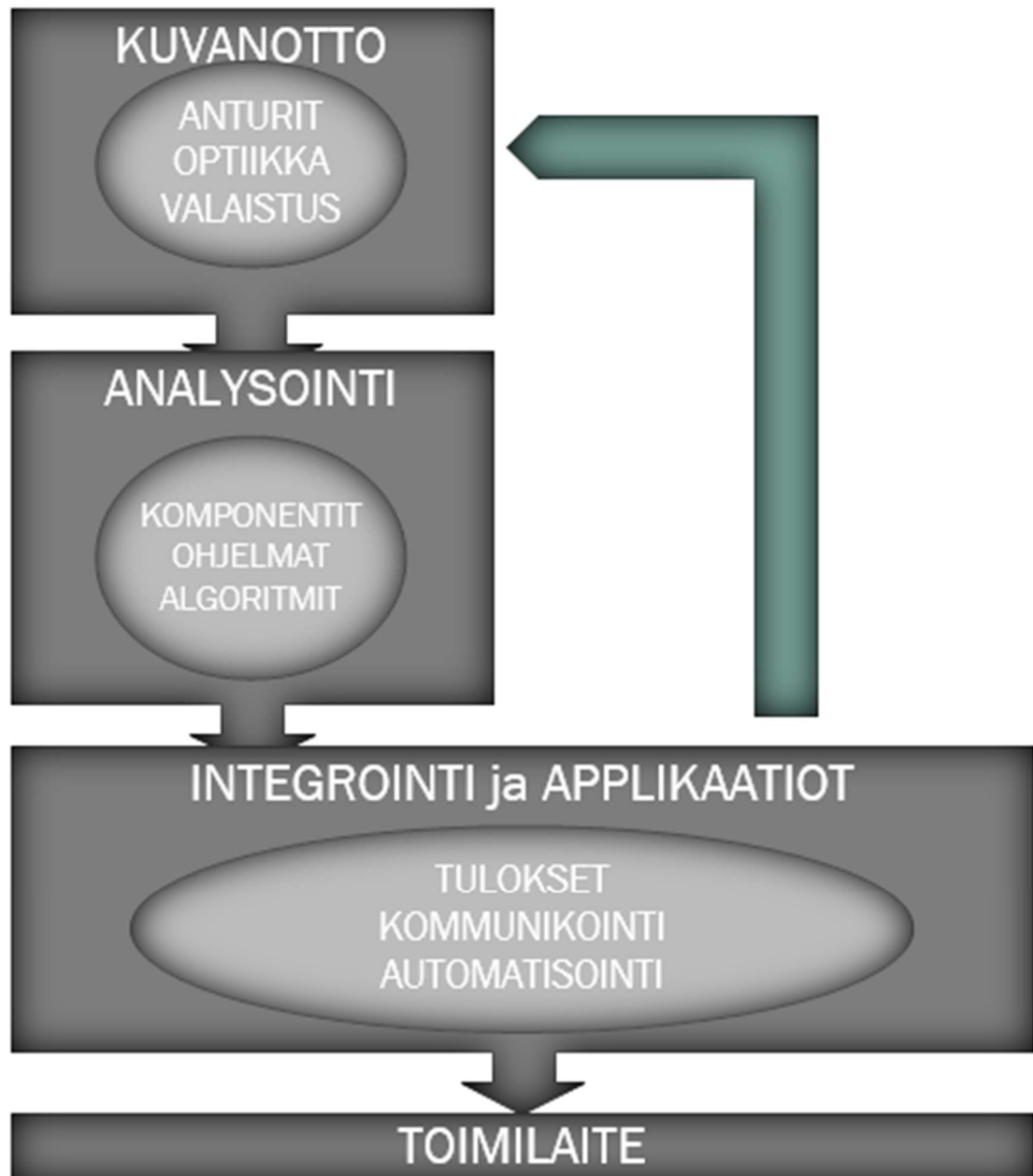
Laitteiden kirjo aiheuttaa lisää asiantuntemuksen tarvetta myös laitteiden tilaajille ja suunnittelijoille, joille eri menetelmien ja järjestelmien ominaisuudet, päivitettävyyden ja ylläpidon kustannukset voivat aiheuttaa yllättäviä lisäkustannuksia tai sellaisen järjestelmän hankinnan, jonka resurssit ylittyvät tarpeettomasti. Fyysisten komponenttien lisäksi markkinoilta löytyy vastaavasti valtava määrä valmiita ohjelmistoja ja kirjastoja, joiden tulkinta ja yhteensopivuus keskenään ja komponenttien kanssa vaatii niin suunnittelijoilta kuin suorittavan tason henkilöiltä laajaa alaan liittyvän teknologian ymmärtämistä ja ohjelmoinnin hallintaa.

Applikaatiokohtaisesti tarvittavien konenäköjärjestelmien käyttöaste voi myös vaihdella jatkuvista, sadasosasekuntien välein tapahtuvasta toistoista, aina hetkelliseen tilannekohtaiseen suoritukseen. Konenäköjärjestelmä määrittely vapaasti suomennettuna [1]:

Ihmisen visuaalisen havainnon ja siitä syntyvän päättelyn korvaaminen kuvaa tallentavalla tai tulkitsevalla laitteella, joka suorittaa halutusta kohteesta tarkastuksen ja saadun tiedon perusteella suorittaa pyydetyn toiminnon.

Valmiita toimittajien hyllyiltä hankittavia järjestelmiä toimitetaan ohjelmistoinen, ja niissä halutun datan, tässä tapauksessa kuvatulkinnan prosessointi on integroituna kameraan tai anturiin. Data lähetetään yleisimmillä väylätekniikalle (CAN) tai teollisuudelle kehitetyillä IE tiedonsiirtoprotokollilla. Laajemmat järjestelmät koostuvat useista eri komponenteista, jopa eri valmistajien komponenteista, ja järjestelmää suunnittelevan tahon näkemyksestä sekä asiakkaan tarpeista. Yhteisiä tekijöitä

konenäköä vaativissa prosesseissa ovat automatisointi, kuvan hankkiminen, analysointi ja datan jatkojalostus ohjaustoiminnoksi (kuva 1). Tässä teoriaosuudessa käydään läpi yleisimpiä kuvausmenetelmiä, komponentteja ja niiden mahdollisia käyttökohteita. [1; 3.]









Kuva 1. Konenäköjärjestelmän arkkitehtuurin havainnekuva.

2.1 Konenäkö

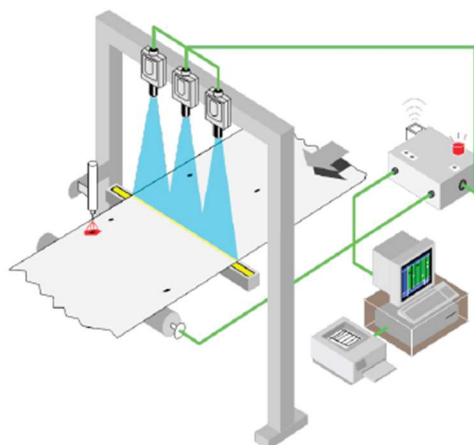
Konenäköä voidaan helposti jakaa vielä useampaan alaryhmään. Tarkastelemme tässä teollisiin tarkoituksiin kehitettyä kuvanottoa teollisuuskameralla sekä 2D- ja 1D-järjestelmillä. Eroa näiden kesken on teollisuuskameran rajatumpi käyttötarkoitus, joka on yksinkertaisten tarkastuksien suoritusta, esimerkiksi tuotepakkauksien laadunvalvonta tai läsnäolon todentaminen. 2D-järjestelmillä sen sijaan voidaan suorittaa mittauksia, koodinlukua, ohjausta, OCR-havainnointia ja OCV-todentamista (kuva 2).

Cognex comprehensive line of vision sensors, 2D vision systems and 3D laser profilers all use machine vision technology to perform inspections but are engineered for different tasks.

	Vision Sensors	2D Vision Systems	3D Laser Profilers
 Presence/Absence	✓	✓	✓
 Inspection	✓	✓	✓
 Guide/Align		✓	✓
 OCR/OCV		✓	✓
 Code Reading		✓	
 Gauge/Measure		✓	✓

Kuva 2. Cognex tuoteryhmät erilaisille konenäköä vaativille sovelluksille [2].

1D-järjestelmä vastaa tavallaan 2D-järjestelmää, sillä erolla, että luettavan kohteen kuvaaminen vaatii liikettä anturia tai kohdetta liikuttamalla. 1D-anturi lukee pinnasta yhden viivan leveydeltä kerralla ja yhdistää tallennetut viivat tulkittavan kuvan luomiseksi (kuva 3). Etu 1D-järjestelmällä on kyky tulkita kuvaa sellaisten kappaleiden pinnoilta, joita ei voida tulkita luotettavasti 2D-tasossa, esimerkiksi koodin luku pyöreän pinnan päältä. [3, s.17.]



Kuva 3. 1D-järjestelmä skannaa liikkuvaa kohdetta rivin leveydeltä kerrallaan [3].

Järjestelmistä löytyy valmistajasta riippuen useita variaatioita, integroitua tuotteita joiden mukana toimitetaan PC:lle ladattavat ohjelmointityökalut tai räätälöityjä, usean eri komponentin kustomoituja ratkaisuja, asiakkaan tarpeen mukaan. Konenäköjärjestelmään kuuluu digitaalinen anturi, joka on integroitu teollisuuskameran sisälle, applikaation vaatimuksien mukainen valaisinyksikkö ja erikoisoptiikka. Valitulla kokoonpanolla on pystyttävä mittaamaan ja tallentamaan kohteesta tarvittavat piirteet analysointia ja datan jatkokäsittelyä varten, joka voi tapahtua joko kamera-anturijärjestelmässä tai erillisellä prosessorilla. Integroiduissa järjestelmissä on useimmiten mahdollista saada valmiiksi prosessoitu data ulos, hyödyntämällä valmiita laiteohjelmapaketteja (Firmware). Ohjelmilla välittää mm. valmiiksi laskettuja asematietoja, nopeustietoja, kyllä tai ei-hyväksyttyä tarkastusta jne. [1; 3.]

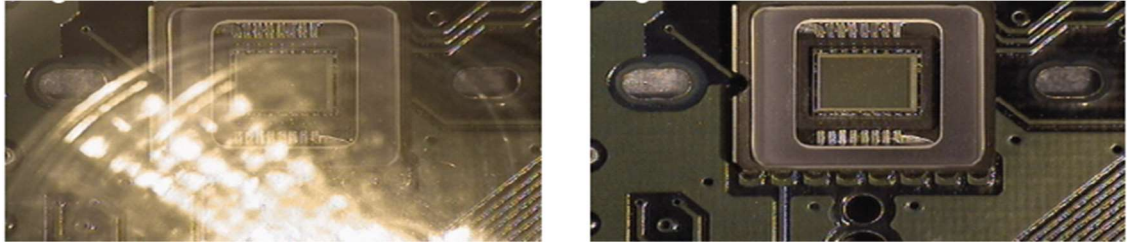
Konenäköjärjestelmän luotettavan toiminnan tärkein tekijä on valaistuksen laatu, jonka tulisi pysyä mahdollisimman tasaisena ja muuttumattomana kalibrointien ja käyttöönoton jälkeen. Valaisuun käytetään kohteesta riippuen useita menetelmiä, kuten taustavalotusta, suunnattua valoa liikkumattomalle kohteelle ja strobovaloa eli pulssivaloa, jolla saadaan liikkuvasta kohteesta tarkka kuva ilman sumentumia. Valintaan vaikuttavat ympäristön valaistus, kohde ja kohteesta halutut piirteet. Pääsääntöisesti mitä monimutkaisempaa kuvaa joudutaan analysoimaan, sitä tärkeämmäksi valotus muuttuu. [4; 5.]

Konenäön valaistuksen neljä kulmakiveä ovat

- geometria - kohteen, valon ja kameran välinen 3D-avaruus
- struktuuri tai kuvio – valon muoto joka projisoidaan kohteelle
- aallonpituus tai väri – miten valo heijastuu tai imeytyy kohteeseen ja taustaan
- suodattimet – miten estetään tai läpäistään aallonpituuksia ja/tai valon suuntaa.

Valolähteinä käytetään yleisimmin LED: iä, loisteputkia, halogeeni- ja Xenon-valoa. Kohteen valaisuun, tekniikkaan, testaamiseen ja valintaan kannattaa uhrata aikaa, loppujen lopuksi koko kuvauksen ja kuvakaappauksen tulkinnan onnistuminen on valosta syntyvien kuvioiden kontrastien tulkitsemista. Ympäristön aiheuttama valaistus (*Ambient Light*) on joissakin tapauksissa suurin haaste konenäköjärjestelmän luotettavan toiminnan kannalta. Tila, jossa kuvausta suoritetaan, olisi pystyttävä pitämään mahdollisimman homogeenisena valoisuuden suhteen. Myös kohteen pinnan heijastavuudesta aiheutuu kuvauslinssiin takaisin suuntautuvia, joskus spektrisiä heijastuksia tai loistetta joka on poistettava optiikan yhteydessä käytettävillä suodattimilla. [4; 5.]

Tyypillisimmät suodattimet optiikan yhteydessä ovat väriinssit (*Colored Glass Filter*), erittäin yksinkertainen tekniikka jolla on mahdollista parantaa kuvalaatus huomattavasti, kaventamalla anturin havaitsemia aallonpituuksia. Väritön tiheydensuodatus (*Neutral Density Filter*) on tekniikkaa, jonka avulla vaikutetaan anturille läpäisevän kirkkauden määrään, esim. hitsauskohteissa tai muissa kohteissa, jossa on voimakasta valoa ja pääsääntöisesti sitä käytetään kirkkauden suodattamiseksi alaspäin. Polarisointisuodattimilla (*Polarization Filters*) suodatetaan peilimäisiä heijastuksia. Polarisointisuodatuksessa käytetään sekä valolähteelle että optiikalle omia suodattimiaan parina, joita nimetään polarisoijaksi (*Polarizer*) ja analysaattori (*Analyzer*) (Kuva 4). [5; 6; 7.]



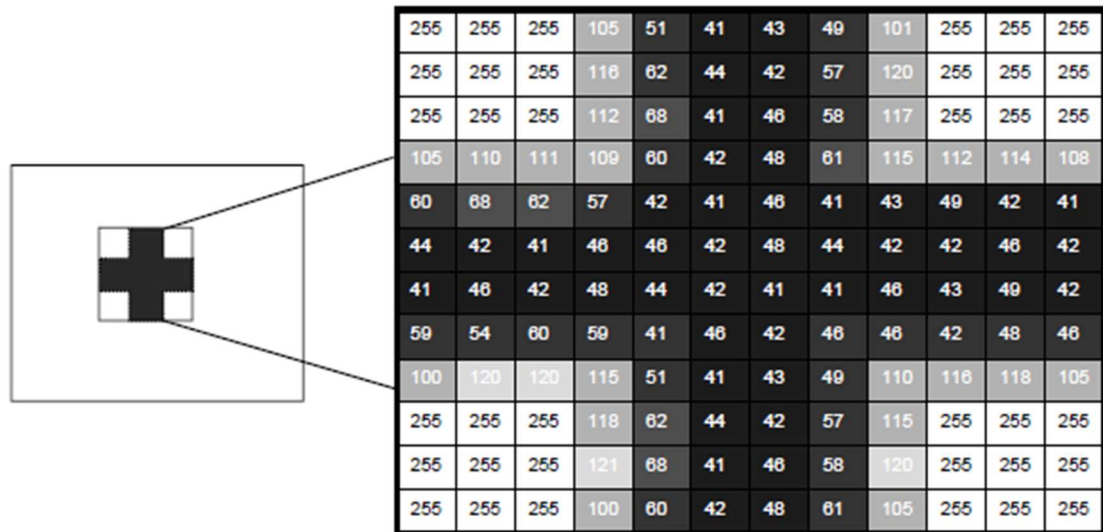
Kuva 4. Vasemmassa kuvassa ei ole lainkaan suodatusta käytössä, oikean puoleisessa kuvassa käytetään polarisointisuodatusta [7].

2.2 Kuvankäsittely (Image Processing)

Kuvakaappaukset manipuloidaan useilla eri menetelmillä, minkä jälkeen tulkitaan haluttuja piirteitä. Näihin tarkoituksiin kehitetyillä ohjelmilla luodaan kuvasta pikselimatriisi, joka toimii pohjana piirteitä tulkitseville sovelluksille. Kuvanottotilanteessa valoisuus tai muu olosuhde saattaa kohteessa poiketa epäsäännöllisesti, jolloin kontrastiarvot poikkeavat toisistaan muuten kahden vastaavan kappaleen välillä. Erilaisilla kuvaa terävöittäville ja segmentointia suorittavilla korjauksilla saadaan kuvista tasalaatuisempia ja luotettavasti tutkittavia.

Riippumatta kuvanottotekniikasta kuva tallennetaan matriisiin, joka koostuu pikseleistä X ja Y koordinaatistossa 2D-tasossa ja lisäksi Z-koordinaatista, kun kyseessä on 3D-kuva. Pikseleillä on tietty intensiteetti tai kontrastiarvo. Tutkittavalle kuvalle suoritetaan tyypillisesti useita kuvankäsittelyprosesseja, joiden pohjalta tarvittava data saadaan eriteltyä ei-tarpeelliselta tiedolta. Prosesseihin kuuluu kuvan perussuodattaminen, jonka jälkeen suoritetaan jokaiselle pikselille muunnos, sen aikaansaaman valon intensiteetin perusteella.

Muunnoksessa määritellään harmaat sävyt (*Greyscale Value*) valkoisesta, joka on voimakkain, mustaan joka on heikoin (kuva 5).



Kuva 5. Havainnekuvana matriisi, jossa kullekin pikselille on luotuna harmaan sävyn (*Greyscale Value*) arvo, joka määräytyy kuvanotto hetken kontrastien tulkinnessa [1].

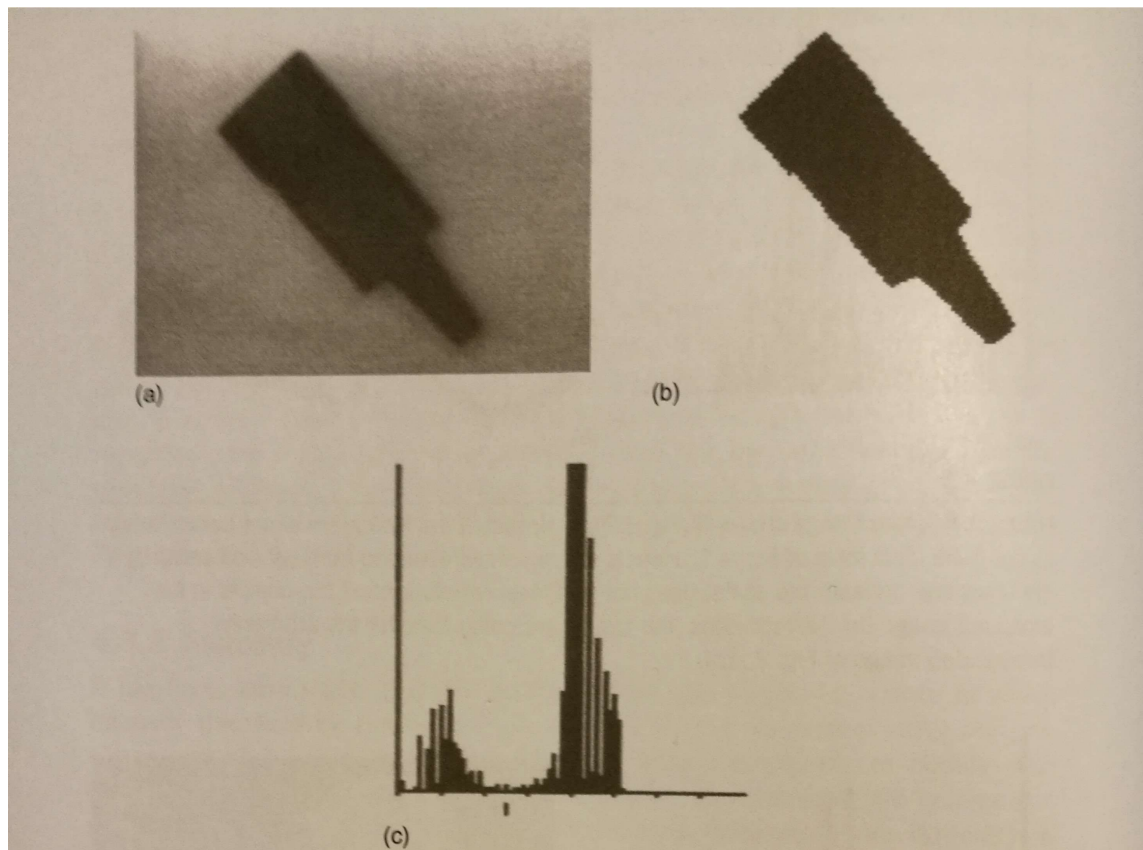
Sävyistä luodaan taulukko, jossa ne esitetään bittiarvoina väliltä 0–255. Kuvamatriisista luodaan tämän jälkeen histogrammi, joka voidaan kuvailla XY-tasossa, niin että X-akselilla on kaikki mahdolliset harmaat sävyt ja Y-akselilla nähdään kunkin harmaan sävyn määrä pikseleitä, joita kuvasta on havaittu. Histogrammin tietoa voidaan hyödyntää mm. tietylle kynnyksarvolle (*Threshold*) muunnoksille, jossa tietyn arvon alittavat arvot muutetaan valkoiseksi ja ylittävät arvot mustaksi. [1; 3.]

Syntyneillä arvoilla, erotellaan kuvasta eri alueita ja osia sitä seuraaville prosesseille. Syntyneestä taulukosta voidaan sitten suorittaa analysointia, pikseleiden tummien ja vaaleiden määristä, segmentoida kuva pienempiin osiin, suorittaa reunojen tunnistusta ja erilaisten kuvioiden ja symbolien tunnistusta. [1; 3.]

Haluttujen piirteiden (*Image Features*) tarkistamista varten on kehitelty algoritmeja, joilla voidaan tutkia ja analysoida rajattuja alueita kuvasta. Tarkastellaan seuraavaksi kynnyksarvon muuttamista tai toisin sanoen kynnyksvertailua (*Thresholding*), joka on yksi käytetyimmistä tavoista suorittaa kuvan segmentointia, ja muutamia tyypillisimpiä perusalgoritmeja, joita ovat reunantunnistaminen (*Edge Detection*), tietynlaisten alueiden tunnistaminen (*Blob Detection*), ja hahmon tunnistaminen (*Pattern Matching*). [8.]

2.2.1 Kynnysvertailu (Thresholding)

Kynnysvertailu (*Thresholding*) on yksi tehokkaimmista ja yksinkertaisimmista tavoista segmentoida kuvaa. Algoritmeja on kehitelty useampia kyseiselle operaatiolle riippuen kohteen vaativuudesta. Yhteinen tekijä on havaita kuvasta halutut pääpiirteet tunnistamalla väreistä tai sävyistä raja, toisin sanoen kynnysarvo jonka ylittävät ja alittavat intensiteetit muunnetaan bittikartaksi, jossa on valkoinen ja musta väri (kuva 6). Näin saadaan prosessi kuvan tulkitsemiseksi huomattavasti helpommaksi, kuin useiden värien samanaikainen tulkinta. Kynnysarvon muuttaminen on kuvan analysoimiseksi helppo ja nopea tapa, jota käytetään laajasti teollisuudenaloilla, joilla on helpot kuvauskohteet ja olosuhteet esimerkiksi jonkun tuotteen todentamiseksi. [1; 8, s. 83.]



Kuva 6. Thresholding, värien kynnysarvon muuttaminen kuvasta (a) haluttujen piirteiden tulkitsemiseksi (b) ja histogrammi (c) josta kynnyksen raja lasketaan vaaleiden ja tummien pikseleiden väliltä [8, s. 90].

2.2.2 Reunantunnistus (Edge Detection)

Reunantunnistuksessa etsitään kuvasta rajoja, joissa pikseleiden sävyt muuttuvat voimakkuudeltaan äkillisesti, joko kohteen fyysisten rajojen tai muun epäjatkuvuuskohdan aiheuttamana. Reunantunnistuksen tarkoitus on löytää kohteelta rajat, joiden kohdalta on mahdollista irrottaa haluttu segmentti ja terävöittää tutkittavaa aluetta kuvasta ja vähentää ei relevanttia tiedon käsittelyä. Yleisimpiä piirteitä, joita reunan tunnistuksessa etsitään ovat suorat tai kaarevat reunat, käytettävästä operaattorista riippuen, tietyissä tai kaikissa suunnissa. Reunan tunnistuksessa käytettäviä yleisimpiä operaattoreita ovat: *Prewitt Operator*, *Sobel Operator*, *Robinson Compass Masks*, *Kircsh Compass Masks* ja *Laplacian Operator*. Menetelmänä reunantunnistus on vaativampi segmentointitapa kuin thresholding [8, s. 111-114.]

2.2.3 Blob-tunnistus (Blob Detection)

Metodissa pyritään löytämään kuvasta pisteitä tai alueita joiden arvot ovat samanlaisia (*Blob*, *Binary Large Object*) tai lähes samanlaisia keskenään ja samalla poikkeavat ympäristöstään, esimerkiksi värin tai kirkkauden perusteella. Tunnistuksella on mahdollisuus hankkia kuvatietoa, joka ei ilmene muilla menetelmillä kuten reunantunnistuksella tai kulman tunnistuksella. Kehitystyön alkuvaiheissa Blob-tunnistusta käytettiin irrottamaan kuvasta tiettyjä alueita (*Regions of interest*) edelleen tutkittavaksi ja prosessoitavaksi. Nykyään käyttö on laajentunut mm. tekstuurien tunnistamiseen ja analysointiin. Viimeaikainen kehitys on laajentanut käyttöä edelleen välittämään informatiivista tietoa kohteen piirteistä ulkomuotoon perustuvassa tunnistamisessa (*Appearance-Based Object Recognition*) ja harjanteiden tunnistamiseen (*Ridge Detection*) esimerkiksi venytetty kappale. [9.]

2.2.4 Hahmontunnistus (Pattern Recognition)

Hahmojen ja kuvioiden tunnistaminen tietokoneella luotettavasti on erittäin haastavaa. Tätä menetelmää edeltää useita muita mm. edellä mainittuja muita prosesseja, ennen kuin kuva on pohjustettu sellaisella tarkkuudella, että tunnistus onnistuu. Kuvioiden tai kuvan tunnistuksessa toisin kuin muissa kuvankäsittelyvaiheissa ollaan kiinnostuneita kokonaisuuksien erilaisista tuntomerkeistä ja klassifioinnista (kuva 7).



Kuva 7. Hahmontunnistus jossa etsitään keksistä tietyttä kuviota [8, s. 365].

Muut menetelmät ja operaatiot sen sijaan suorittavat puhtaasti mittauksia ja matemaattisiin laskelmiin perustuvaa piirteiden tutkimista. Kuvan tunnistusta edeltävissä vaiheissa suoritetaan tarvittavat suodatukset kohinan ja tarpeettomien mittauksien poistamiseksi, segmentoidaan kuva yksilöllisiksi objekteiksi, suoritetaan mittaukset jne. Tuloksista laaditaan sarja eri piirteiden mitoista ja näiden keskinäisistä suhteista, joista luodaan kuvion muoto. Lopuksi verrataan syntynyttä muotoa sitä vastaavaan tallennettuun kuvioon tai kuvioiden sarjaan. Hahmontunnistuksen alle sijoittuu useita operaattoreita ja analyysimalleja kuten objektin tunnistus (*Object Detection*), muodon analysointi (*Shape Analysis*), pyöreyyden/terävyyden analysointi (*Roundness/Sharpness Analysis*), orientaation analysointi (*Orientation Analysis*) ja asetelman analysointi (*Arrangement Analysis*). [10, s. 2-4.]

2.3 3D-kuvaus ja Time of Flight kamera

2.3.1 3D-kuvauksen perusteita

Kolmiulotteiseen kuvakaappaukseen ja tallentamiseen käytetään laserskannausta, ultraäänikeilausta, eri tyyppisiä 3D-antureita tai yhdistettyjä 2D/3D-älyantureita, joiden kuvanotto perustuu valon tai muun säteilyyn kuluvan ajan mittaamiseen. Laserskannauksessa kohteen pinnalle suunnataan laservaloa, joka skannaa näkyvän pinnan kauttaaltaan systemaattisesti, luoden mitatuista pisteistä pistepilven

kolmiulotteisessa XYZ-koordinaatistossa, joka vastaa kohteen muotoja. Pistepilvi vastaa 2D-kuvauksen matriisia, mutta lisänä on etäisyystieto Z. Laserskannaus on todettu hyväksi menetelmäksi mm. liikkuvilla ja orgaanisilla kohteilla.

Toisena menetelmänä laserskannauksessa käytetään strukturoitua, nk. valkoista tai sinistä valoa, joka lähettää kuvioita kohteen pinnalle ja mittaa takaisin heijastuvien kuvioden vääristymiä. Kyseinen menetelmä sopii hyvin kohteille jotka eivät liiku. Mittausmenetelmä 3D-kuvaus sopii erinomaisesti tehtäviin, joissa kohdetta ei voida jostain syystä fyysisesti koskettamalla mitata tai se on kooltaan erittäin iso, esimerkiksi iso avara tila, jossa on monimutkainen geometrinen toteutus tai epäsäännöllisiä pintoja [8, s. 389–392; 12.]

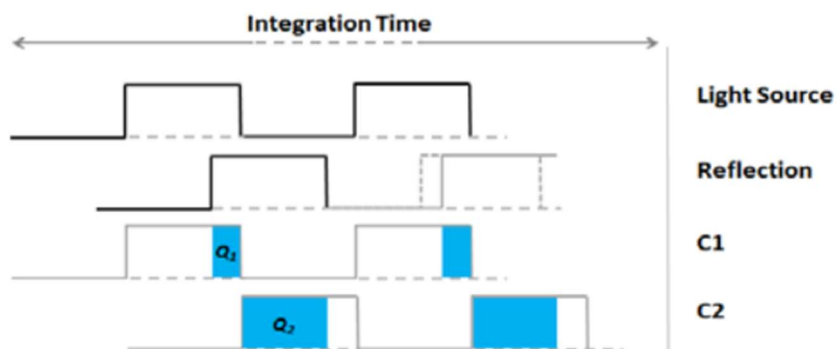
Tässä projektissa hyödynnettiin IFM O3M 251 Smart Sensoria (kuva 8). Laite mittaa optisella järjestelmällä kohteen pinnan ja itsensä välistä etäisyyttä nk. PMD-tekniikalla. O3M-anturit ovat ns. uuden sukupolven ToF-kameroita (*Time-of-Flight Kamera*) jotka on kehitelty ajoneuvoille ja muille liikkuvan kaluston mobiilisovelluksille. Valaisinyksikkö lähettää pulsseina infrapunavaloa synkroniassa optisen laitteen kanssa, joka mittaa heijastukseen kuluvan ajan. Tekniikalla saadaan hyvin tarkkoja etäisyyksiä reaaliajassa. Pääkomponenttina on rivi- tai matriisianturi joka mittaa etäisyyden jokaiselle pikselille samanaikaisesti rinnakkain. Tuloksena on nopea kuvanottotekniikka, jolla on korkeatasoinen lateraalinen resoluutio ja syvyytietä.

Projektissa käytetyn O3M-sarjan kameroilla on 1 024 (64*16) pikselin resoluutio. PDM-tekniikan kulmakivenä pidetty PDM Technologies on kehittänyt vaativimmille sovelluksille IFM:lle myös O3D-sarjan kameran, 23 000 pikselin resoluutiolla, näytteenotto 20 Hz:n taajuudella, joka toimii suorassa auringon valossa 10 000 luksiin saakka. Molemmilla järjestelmillä 3D-data prosessoidaan anturilla ja luetaan Ethernetin välityksellä tai CAN-väylältä J1939 CAN-viesteistä.



Kuva 8. Projektissa käytetty IFM O3M251 Smart Sensor ja O3M950-valaisinyksikkö.

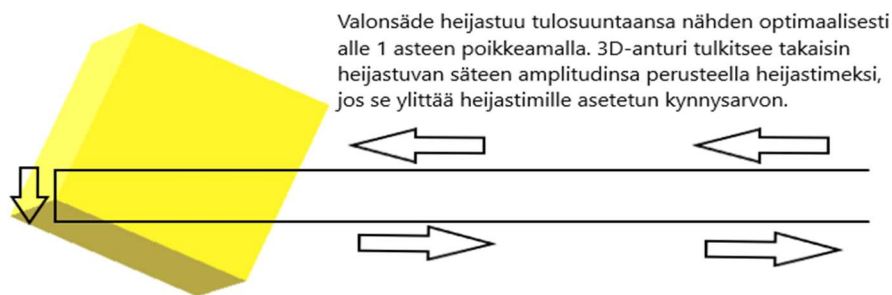
PMD-tekniikalla toimivat kamerat (anturit) ovat halpoja ja kustannustehokkaita ratkaisuja sovelluksissa, joissa riittää matala resoluutio. Kolmiulotteinen kuvanotto onnistuu ilman liikkuvia osia, monimutkaista elektroniikkaa ja datan prosessointia verrattuna laserskanneriin, joka keilaa kauttaaltaan kohteen, ja jonka pohjalta suoritetaan raskaita matemaattisia operaatioita kuvan tulkitsemiseksi. Kuvausmenetelmässä valaistaan moduloituilla valopulsseilla koko havaintokenttä, jonka takaisin heijastuvaa viivettä mitataan jokaisella pikselillä matriisissa erikseen yhdellä kertaa. (kuva 9.)



Kuva 9. PMD-tekнологiaan perustuvaa heijastaman mittausta [12].

Järjestelmän hyöty verrattuna perinteiseen konenäköjärjestelmään on sen riippumattomuus yleisistä valaistusolosuhteista. IFM:n 3D-antureiden järjestelmissä ovat myös huomioituna käyttöolosuhteet ja ympäristön saasteet, joita voidaan konfiguroinnin yhteydessä määritellä käyttöliittymästä, kuten mahdolliset likaisuusasteet, joiden kanssa optiikka saattaa joutua kosketuksiin, ja poikkeavat tilanteet kuten ilman epäpuhtaudet, sumu, savu, pöly jne.

Projektin 3D-anturin päätehtävä on havaita heijastimiksi valmistettuja kappaleita. Heijastimen kyky palauttaa siihen kohdistuvan valonsäde, mahdollisimman pienellä hajonnalla ja kulmalla on perusedellytys heijastintyyppin sopivuudelle tehtävään. Kuvassa 11 on kuvattuna prismaattisen heijastimen valonsäteen ohjaus, suorakulmaisten pintojen kautta, jolla palautetaan säde tulosuuntaansa rinnakkaisena tulosäteeseen nähden. [15.]



Kuva 11. Prismaattisen heijastimen toimintaperiaate.

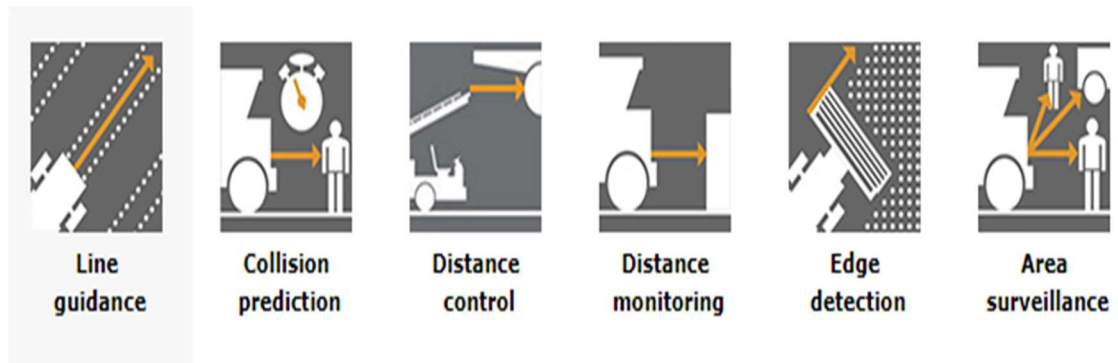
Tehtävään valitulta heijastimelta on oltava alle yhden asteen poikkeama palautuvalle säteelle. Mahdollisimman syvät, prismaattiset ja isokokoiset heijastimet takaavat luotettavan toiminnan myös pidemmillä etäisyyksillä [15].

2.3.2 3D älyanturin käyttö ja mahdolliset sovellukset

IFM O3M-sarjan kamerat on suunniteltu vaativiin olosuhteisiin ja kohteisiin teollisuudessa ja erilaisille mobiilisovelluksille. Projektissa käytetyllä mallilla on lisäksi videokuva, jota voidaan hyödyntää ajoneuvoissa, joissa on tarve saada visuaalinen yhteys jollekin kuljettajan katveeseen jäävälle alueelle mm. peruuttaessa tai ajoneuvon yläpuolella olevalle kohteelle. Erilaisille sovelluskohteille löytyy valmiita, pitkälle kehiteltyjä visuaalisia työkaluja operaattoreiden tueksi näyttömoduulin välityksellä. Ajoneuvojen autonomiseen ohjaamiseen on kehitelty apuohjelmia ja demoprojekteja,

joita on helppo hyödyntää sellaisenaan tai osittain soveltaa omien tarpeiden räätälöimiseksi kuten tässä projektissa.

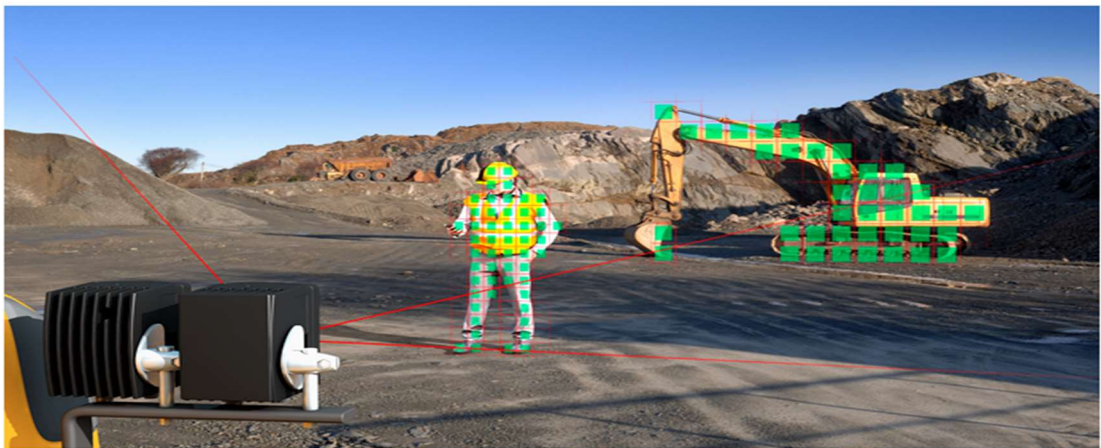
Mobiilisovelluksissa PDM-tekniikkaa pystytään hyödyntämään monipuolisesti (kuva 12) johtuen tarkasta ympäristön ja havaittujen kohteiden (*Objects*) etäisyys- ja syvyystiedosta. Kuvaustekniikalla, jossa 3D-malli tallentuu yhdellä kaappauksella, vältetään liikkeestä aiheutuvaa sumentumaa, joka vaikeuttaa tiedon tulkitsemista.



Kuva 12. IFM O3M 3D-sovelluskohteita [16].

IFM:n PDM-tekniikkaan pohjautuvia älyantureita on sovellettu mm seuraavanlaisiin tarkoituksiin, joko tuotannon autonomiseen ohjaamiseen tai turvallisuus- ja valvontatehtäviin:

- Linjan tai polun seuranta. Anturi vertaa tilavuuden perusteella vierekkäisiä alueita, kuten viljapellolla liikkuvan maatalouskoneen ajolinjaa. Anturi voidaan ohjelmoida antamaan kuljettajalle suosituksia ohjaamiselle tai ohjaustietoa autonomiselle ohjaukselle.
- Törmäyksen ennustaminen ja estäminen. Anturi parametrisoidaan tunnistamaan liikkuvia ja paikallaan olevia kohteita. Ajoneuvon oma ajonopeus ja jarrutukseen kuluvaa aikaa verrataan kohteiden suuntiin ja nopeuksiin. Tieto välitetään kuljettajalle ja ajoneuvon ohjausjärjestelmälle.
- Etäisyyden ja korkeuden valvonta. 3D-kuvassa lasketaan ja välitetään minimi ja maksimi arvot koko kuvatulta kohteelta yhden pisteen sijasta.
- Etäisyyden valvonta. Anturille voidaan ohjelmoida haluttuja etäisyyksiä hälyttämään lähestymisistä ja pysäytyksen tarpeesta.
- Reunantunnistus, jolla on mahdollista ohjata tarkasti mm. leikkuupuimuria pellolla ajamaan mahdollisimman lähellä jo puitua reunaa.
- Alueen valvonta, jossa on mahdollista ohjelmoida erilaisilla luokituksilla olevia kohteita havaittaviksi, esimerkiksi kulunvalvonta jonkin tietyn alueen kohdalla, tai havaitsemaan alueella liikkuvista henkilöistä heijastavissa turvaliiveissä (kuva 13).



Kuva 13. Turvaliivi voidaan luokitella ohjelmallisesti heijastavaksi kohteeksi (*Reflective Object*) ja kuvassa näkyvä kaivinkone liikkuvaksi tai paikallaan olevaksi kohteeksi (*Moving / Static Object*), kun sen heijastusaste jää kynnyksarvon alapuolelle [16].

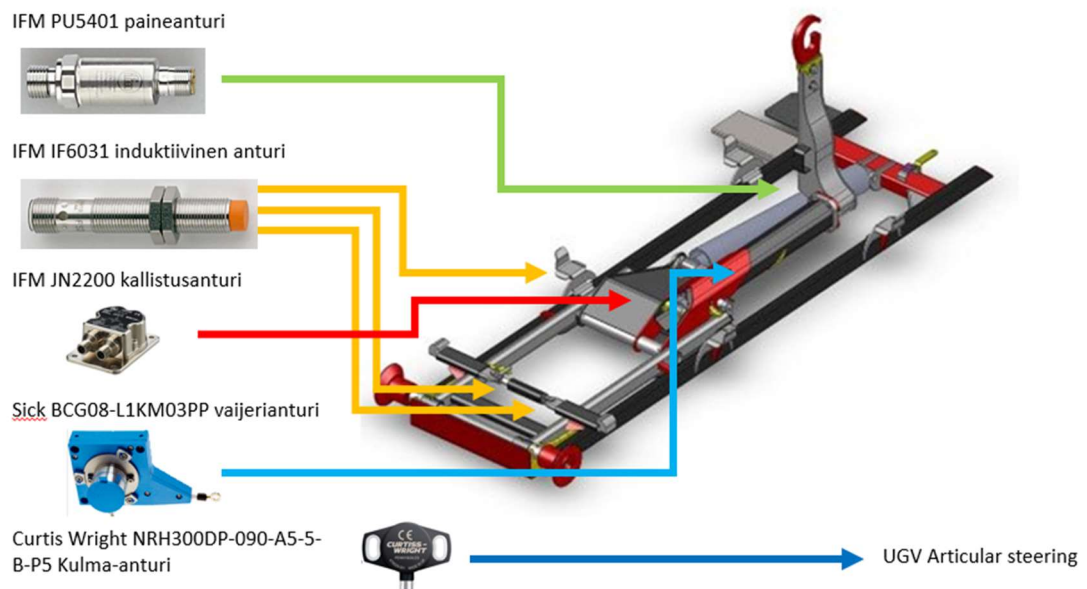
3 Projektissa käytetyt laitteet ja komponentit

Moduulien käsittelylle automatisoitiin italialaisen CPS-Yhtiön Succi SR 3-30-koukkulaite (kuva 14). Koukkulaite on alun perin suunniteltu kevyille kuorma-autoille tai raskaille pick-up-alustoille. Laitteen alkuperäiseen varusteluun ei kuulu anturointia tai sähköisiä turvalaitteita. Hydraulikkajärjestelmä koostuu 3-karaisesta Hydrocontrol Q45-monoblockventtiilistä, kippisylinteristä, liukupalkkisylinteristä ja lukkosylinteristä. Kippi- ja liukupalkkisylinterit on varustettu kuormanlaskuventtiilillä ja lukkosylinteri lukkoventtiilillä.

Koukkulaitteen ja alustan anturointi (kuva 14) on toteutettu seuraavasti:

- IFM PU5401-paineanturi 1 kpl
- IFM IF6031-induktiivinen anturi 3 kpl
- IFM JN2200-kallistusanturi 1 kpl
- Sick BCG08-L1KM03PP-vaijerianturi 1 kpl
- Curtis Wright NRH300DP-090-A5-5-B-P5-kulma-anturi 1 kpl

Alustalla hyödynnettiin LPMS-CU Gyro:n antamaa laitteen kallistustietoa pituussuunnassa.



Kuva 14. Succi SR 3-30 Hook Lift ja automatisointiin käytetyt anturit.

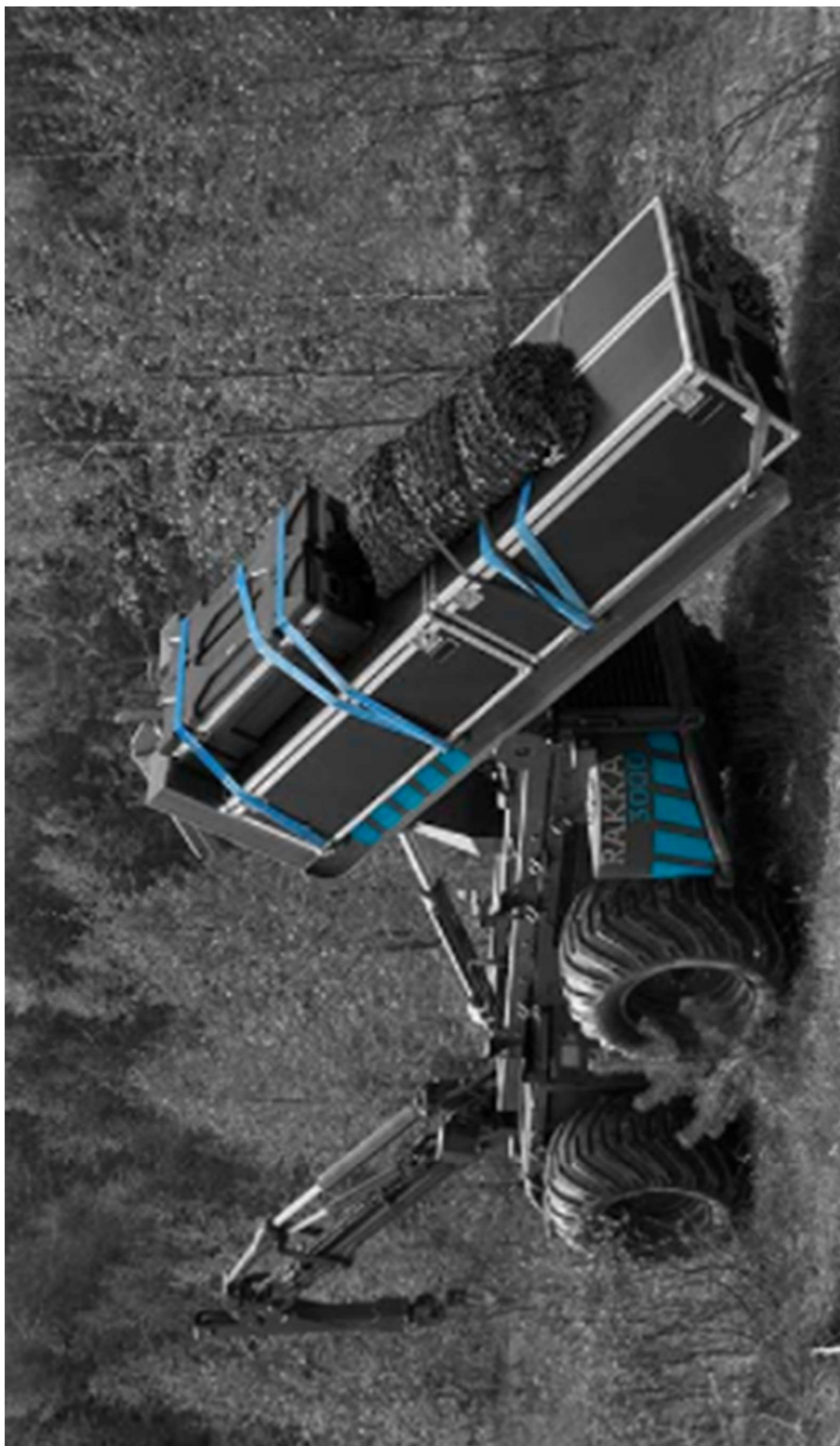
Mekaanisilta osin, laitteessa on liukupalkki, jolla on mahdollista asemoida lavan paikkaa rungon päällä pituussuunnassa 55 cm. Kippirunko on kaksiosainen, keskeltä taittuva, kun liukupalkki on ajettu takimmaiseen asemaansa, lukot avattuna, ja takakippiakselin kohdalta taittuva, kun liukupalkki on ajettu irti taka-asennostaan ja lukot ovat kiinniasennossa. Sivusuuntaiseen lukitukseen käytetään molemmin puolin lavajuoksuja liukuvia lukkokoukkuja, jotka on kiinnitetty toisiinsa kelluvalla sylinterillä.

Moduulinkäsittelylaitteen toimilaitteiden manuaaliseen ja automaattiseen ohjaukseen valittiin Technionin TEC132-logiikkaohjain, ohjelmointi Codesys 2.3-ohjelmointiympäristöllä. Moduulien ja maaston skannaukseen käytettiin IFM O3M 251 Smart Sensoria. Rajapintana O3M 251-anturin, alustan Exertus PLC:n ja moduulinkäsittelyn TEC 132:n välillä, toimi IFM-näyttömoduuli CR 1083, Codesys 2.3-ohjelmointiympäristöllä. (kuva 15)



Kuva 15. Projektin TEC132 PLC, IFM O3M-anturi ja IFM-näyttömoduuli.

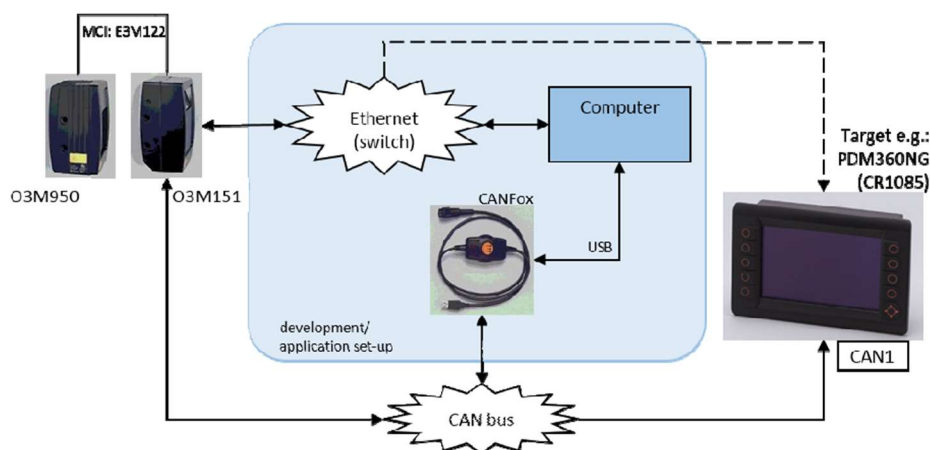
Projektin alustana käytettiin radio-ohjattavaa, Rakka 3000 neljännen sukupolven miehittämätöntä maa-ajoneuvoa, joka on neljällä hydraulisella ajomoottorilla ja kahdella tilavuus-säätöpumpulla varusteltu, dieselkäyttöinen maastoajoneuvo (kuva 16). Alustan automatisointi rajoittuu ajokorkeuden ja eri tyyppisten kellutuksien ohjaukseen. Alustalla ei ole ajonopeutta tai matkaa mittaavia järjestelmiä, eikä myöskään runkonivelen asennon valvontaa, joka aiheutti tässä projektissa omat haasteensa paikoituksen ja siirtymisen nopeuksien hallinnassa.



Kuva 16. Rakka 3000 UGV.

4 IFM O3M 251-kameran ohjelmointi, apuohjelmat ja ohjelmointilaitteet

Kamerajärjestelmän ohjelmoimiseen ja parametrisoimiseen käytettiin IFM Visual Assistant-ohjelmaa versio 1.7.10 (julkaistu 8.9.2017). Ohjelma on ilmaiseksi ladattavissa IFM:n sivuilta, eikä vaadi asennusta vaan ajetaan sovelluksena kansioista. Kamerajärjestelmässä käytetään Firmware OD_4.19.6:tta. Lataus edellyttää Ethernet-kaapelin ja CANfox-kaapelin kytkennän. CANfox on anturin CAN-liittimen ja PC:n USB portin välille kytkettävä adapteri, joka mahdollistaa diagnostiikan ja ohjelmien latauksen kamerajärjestelmälle, CANfox-adapterin käyttö vaatii kyseisen laitteen ajurin asennuksen tietokoneelle. Ajurin voi ladata IFM:n sivulta, mikä vaatii rekisteröitymisen. Hyväksymisen jälkeen IFM lähettää tunnuksen ja salasanan erilliselle lataussivulle. Linkki latauspaikoille on liitteessä 1. Kytkennästä on havainnekuva (Kuva 17). [18.]



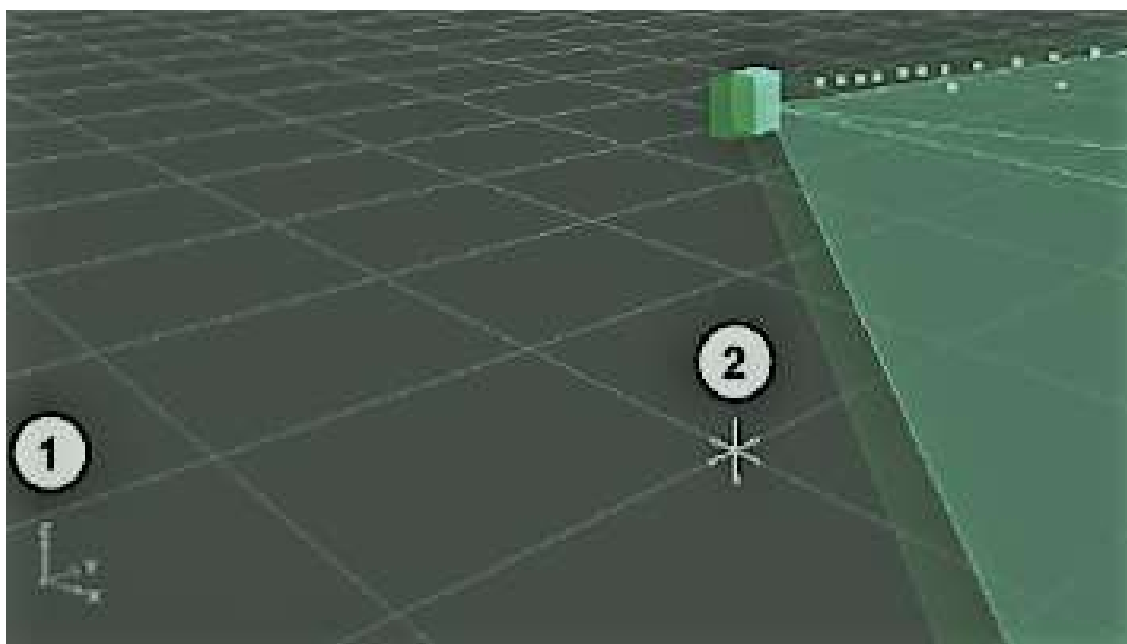
Kuva 17. CANfox sovitin ja muu kytkentä O3M-XXX kameroiden ohjelmointiin [19].

4.1 Visual Assistant

Visual Assistant on työpöytäsovellus, jolla asennetaan anturijärjestelmälle halutut laiteohjain-paketit (*Firmware Packages*), suoritetaan konfigurointi ja parametrisointi. Sovelluksessa olevalla *Device Setup*-toiminnoilla konfiguroidaan ja *General Settings Wizard*-toiminnolla säädetään haluttuja ominaisuuksia asennetulta paketilta. *Calibrations Settings*-toiminnolla suoritetaan anturin asematieto haluttuun maailmankoordinaatistoon ja kalibrointi. Lisäksi on mahdollista tarkastella laitteen asennusta ja muita tietoja sekä nauhoittaa ja toistaa kuvaa. [18.]

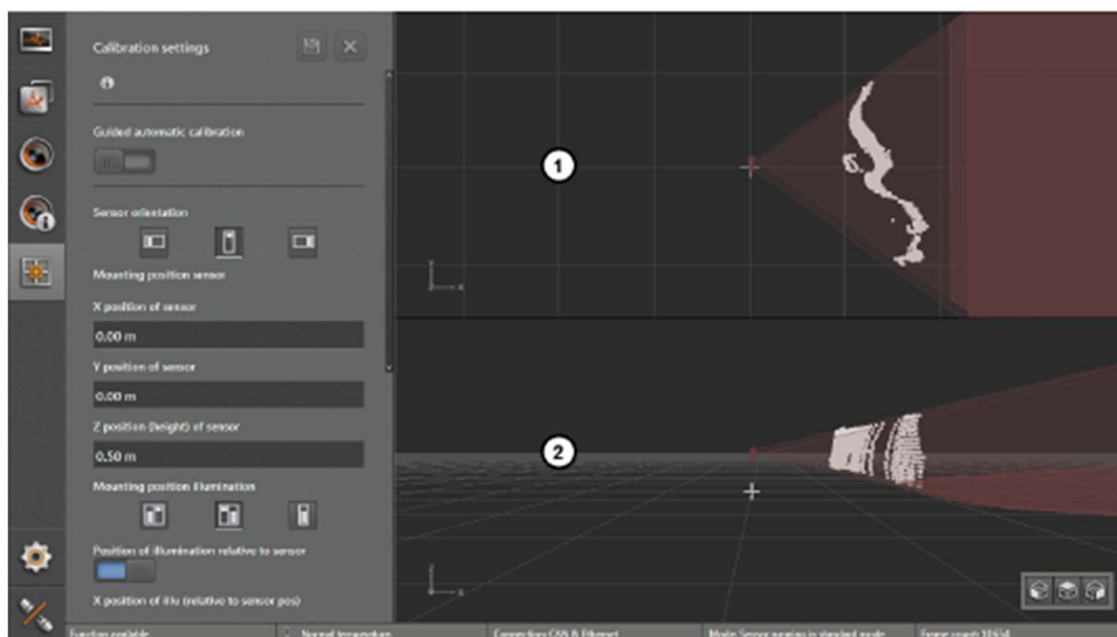
4.2 3D/2D-anturin asennus

Anturilla mitattava 3D-data luetaan maailmankoordinaatiston suhteen, joka voidaan määrittellä haluttuun kohtaan ajoneuvossa tai sen ulkopuolella. Anturin asennuspaikan ja maailmankoordinaatiston suhteen syötetään järjestelmälle tieto manuaalisesti ja voidaan myös hyödyntää automaattista hienosäätötyökalua. Kuvassa 18 nähdään XYZ-akselit (1) ja anturin asema maasta (2) origoon nähden. Asennuspaikasta ja tavasta riippuen on syötettävä mahdollisimman tarkasti referenssipisteeseen mitattuna anturin korkeus Z ja mahdollinen kallistus Y. [18, s. 59–61.]



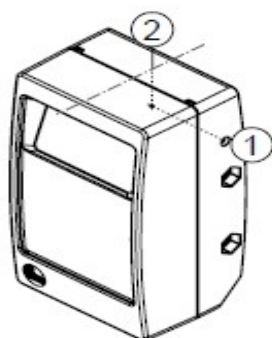
Kuva 18. Anturin paikka suhteessa maailmankoordinaatistoon [18].

Tarkennettu kalibrointi tapahtuu ohjatusti kytkemällä *Guided Automatic Calibration*-valinta, joka vaatii karkean kameran asennuskorkeuden ja kallistuskulman ensimmäisellä kalibroitikerralla (Kuva 19). Anturin kuvausalueella edessä ja molemmilta sivuilla on oltava tyhjää tilaa maatasolla riittävältä alueelta. Automaattinen kalibrointi käyttää maapintaa referenssinä asemansa laskemiseen. Tämä huomioon ottaen olisi laitteiston, johon anturi on asennettu, oltava mahdollisimman suorassa ja lattia- tai maapintojen oltava mahdollisimman tasaiset. [18, s. 65.]



Kuva 19. Anturin paikka- ja asentotieto syötetään varatuille paikoille.

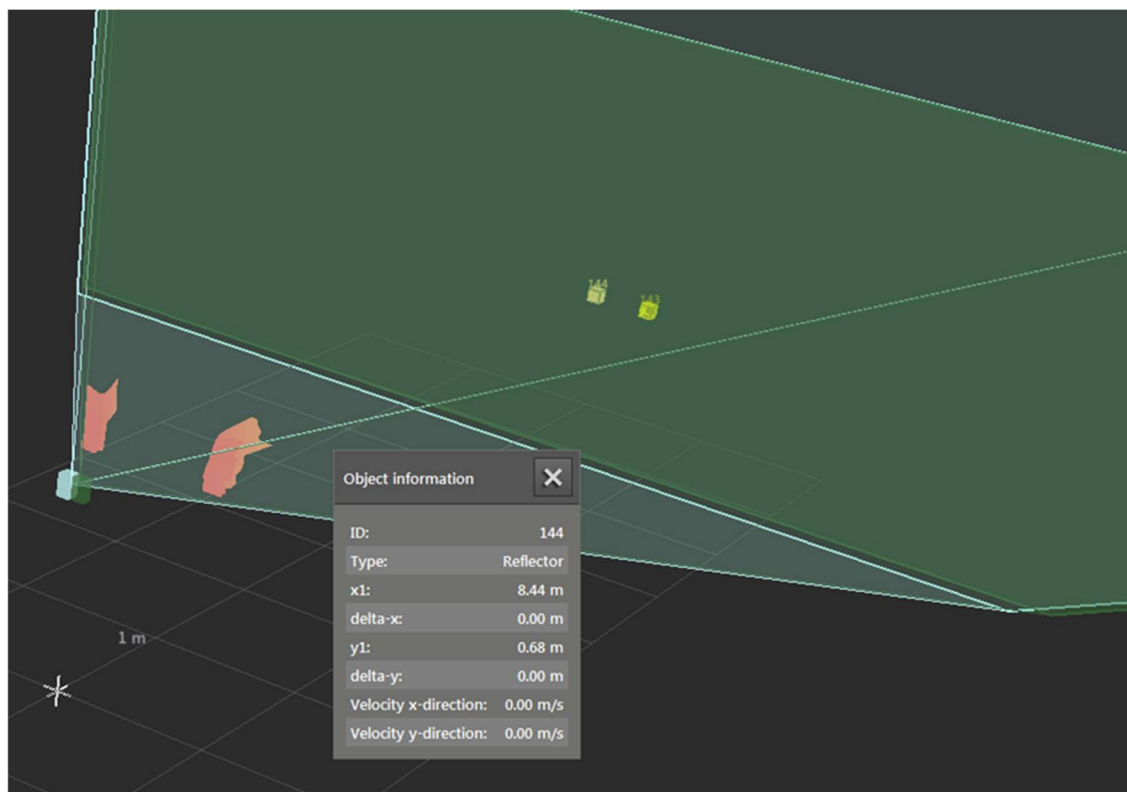
Kameran fyysisellä sijainnilla ja asennolla on ainoastaan muutamia rajoitteita. Valaisinlaitteen ja anturin välinen etäisyys toisistaan voi olla 0–280 cm välillä. Kun järjestelmää käytetään heijastumien havaitsemiseen, niin valaisinlaite tulisi asentaa mahdollisimman lähelle anturia. Minimikorkeutena anturille suositellaan 0,5 m. Anturilla on tarkkaan määritelty referenssipiste, jolla sen paikka saadaan laskettua maailmankoordinaatistoon nähden (kuva 20). [19, s. 6.]



Kuva 20. Anturin referenssipiste on määritelty kohdan (1) uran ja laitteen keskilinjan (2) kohdalle [20].

4.3 Kameran konfigurointi ja parametrisointi

IFM:llä on tarjolla ladattaviksi kameraan erilaisia laiteohjaimia paketteina, joilla on mahdollista räätälöidä järjestelmän ominaisuuksia vastaamaan tarpeita. Tarjolla on OD (*Object Detection*), DI (*Distance Image*) ja LG (*Line Guidance*). Tässä projektissa käytettiin OD pakettia, joka mahdollistaa erilaisten pintojen ja heijastimien havaitsemisen ja näiden koordinaattien laskemisen (kuva 21).



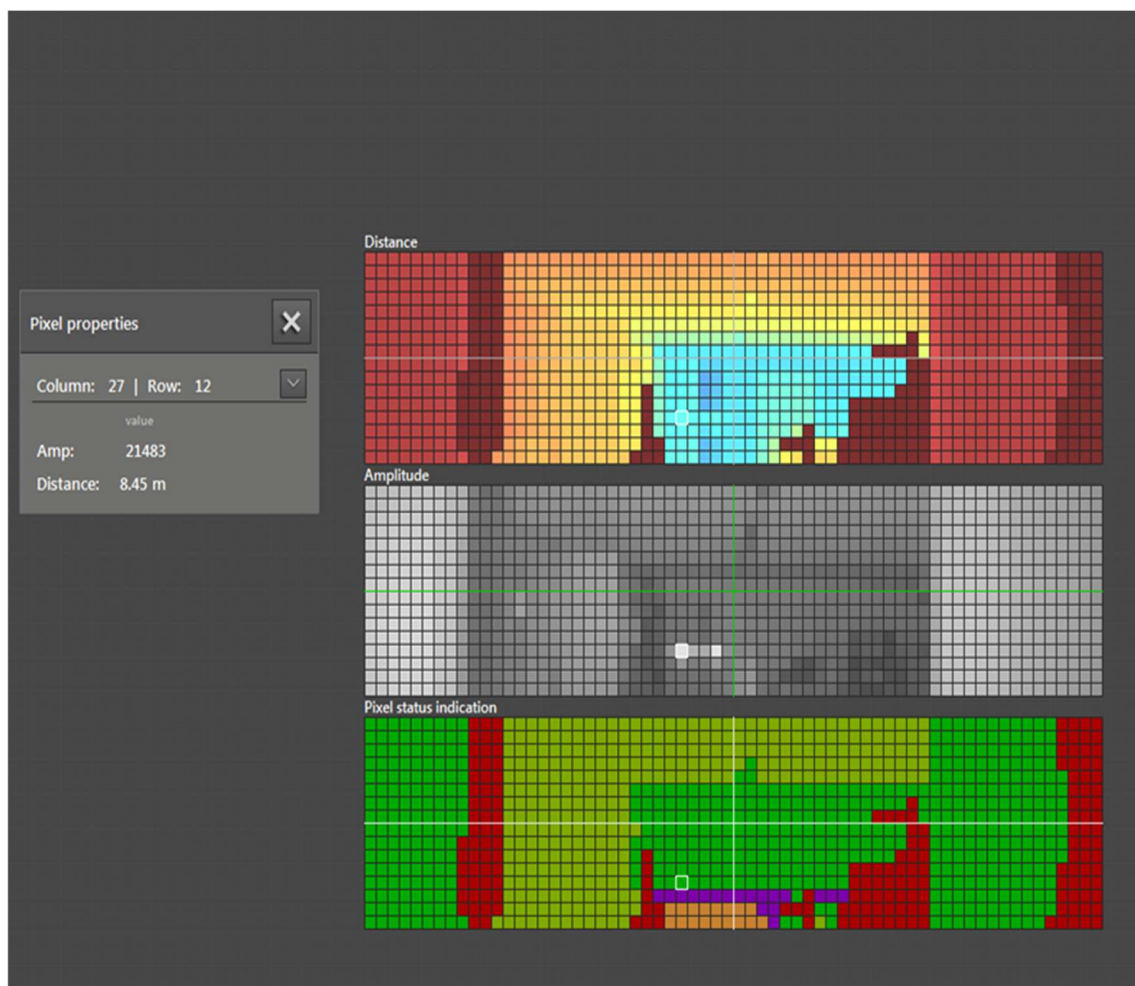
Kuva 21. Visual Assistantilla tarkasteltua 3D-kuvaa. Näkymässä 2 kpl havaittuja heijastimia asennettuna 60 cm:n päähän toisistaan samalle korkeudelle. Kuvassa on valittuna vasemmanpuoleinen objekti, jonka paikkatiedot anturin suhteen tunnisteella ID 144 tyyppiä heijastin (*Reflector*).

Heijastamien ns. kynnyisarvo (*Threshold Value*) säädetään Visual Assistantin avulla tasolle, joka suodattaa pois mahdollisia vääriä havaintoja. Liian matala arvo havaitsee myös tavallisia heijastavia pintoja kuten kiillotetut pinnat, kiiltävät maalipinnat, kromiosat jne.

Kun kynnyisarvo säädetään riittävän korkealle tasolle, poistetaan tehokkaasti vääriä havaintoja (muita satunnaisia heijastumia) ja anturi tunnistaa varsinaisiksi heijastimiksi

valmistettuja kappaleita. Kynnysarvon tasoa kannattaa testata useilla tasoilla ja tarkistaa Visual Assistantin välityksellä heijastimien taajuus ja ympäristön muiden objektien taajuuksia ja arvoja. Kaikki säädöt ovat tarpeen suorittaa olosuhteissa, joissa on voimakas auringonpaiste tai vastaava valaistus, ja myös auringonpaisteessa määrällä ympäristöllä, jossa luonnollisia voimakkaita heijastuksia tulee eri pinnoilta. [20, s. 82.]

Tarkastelemalla näkymää tietokoneen näytön avulla, heijastimet nähdään erillisinä objekteina, joille luodaan ID-numero (kuva 22). Käyttämällä korkeilla taajuuksilla toimivia heijastimia, on mahdollista rajata tehokkaasti pois virrehavaintoja.



Kuva 22. 2D-kuvana tarkasteltuna kohde, heijastavan kohteen amplitudi valitussa pikselissä on 100-kertainen tavallisiin ympäristön pintoihin verrattuna.

5 Kallistusanturin konfigurointi

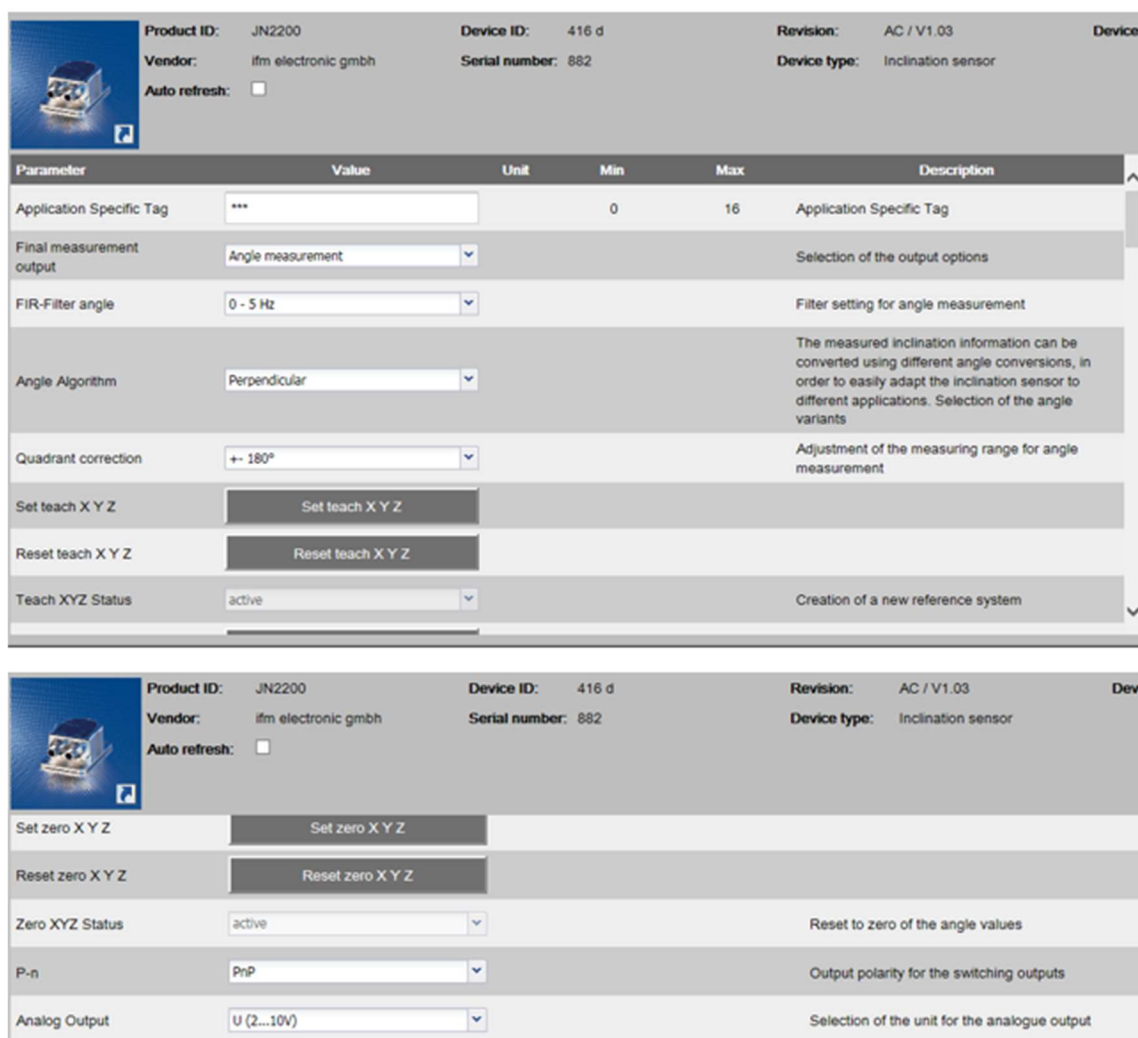
Kippirungon asennon valvontaan on asennettu IFM JN2200-kallistusanturi. Anturin parametrisointi, arvojen ja prosessidatan lukeminen, diagnostiikka ja konfigurointi ovat mahdollista I/O-Link interfacen avulla. Tehtävä vaatii IFM E30390 USB IO-LINK MASTER sovittimen, joka kytketään PC:n USB-portin ja anturin väliin (kuva 23). Laite sopii kaikkien IFM I/O-linkillä varusteltujen antureiden konfigurointiin. Laitteen ajuri (*Driver*), laiteohjain (*Firmware*) ja anturikohtaiset lisäohjelmat ovat ladattavissa tuotteen sivulta (liite 1).



Kuva 23. IFM I/O Link Master ja kippirungon alle asennettu JN 2200 kallistusanturi.

Anturi on konfiguroitu lähettämään 2–10 V:n analogisignaalia. Tässä projektissa hyödynnettiin ainoastaan pituussuunnan kallistusta. Etuna perinteisiin kiertokulman mittaamiseen perustuviin antureihin nähden on helppo ja nopea asennus, mihin kohtaan tahansa kipin runkoa, edullinen hinta ja konfiguroitavissa olevat lisäominaisuudet. Konfiguroinnista ja parametrisoinnista näkyy kuvassa 24 käyttöliittymänäkymä.

Anturin asennuspaikka kippirungossa huomioi sekä kipin asennon että alustan pituussuuntaisen kallistuksen. Kokonaiskulman tieto mahdollistaa puomin asemoinnin korkeuden suhteen, autonomisessa lavan laskemisessa tai nostotilanteessa moduulille optimaaliseksi, riippumatta maastosta.



The image shows two screenshots of the JN2200 sensor configuration software interface. Both screenshots display the same header information: Product ID: JN2200, Device ID: 416 d, Revision: AC / V1.03, Vendor: ifm electronic gmbh, Serial number: 882, and Device type: Inclination sensor. The 'Auto refresh' checkbox is unchecked.

Top Screenshot (General Settings):

Parameter	Value	Unit	Min	Max	Description
Application Specific Tag	***		0	16	Application Specific Tag
Final measurement output	Angle measurement				Selection of the output options
FIR-Filter angle	0 - 5 Hz				Filter setting for angle measurement
Angle Algorithm	Perpendicular				The measured inclination information can be converted using different angle conversions, in order to easily adapt the inclination sensor to different applications. Selection of the angle variants
Quadrant correction	± 180°				Adjustment of the measuring range for angle measurement
Set teach X Y Z	Set teach X Y Z				
Reset teach X Y Z	Reset teach X Y Z				
Teach XYZ Status	active				Creation of a new reference system

Bottom Screenshot (Zeroing and Output Settings):

Set zero X Y Z	Set zero X Y Z				
Reset zero X Y Z	Reset zero X Y Z				
Zero XYZ Status	active				Reset to zero of the angle values
P-n	PnP				Output polarity for the switching outputs
Analog Output	U (2...10V)				Selection of the unit for the analogue output

Kuva 24. JN2200 anturin konfigurointi ja asennuksen jälkeisen asennon nollaaminen käyttöliittymässä.

Asennuksen ja konfiguroinnin jälkeen anturin asema nollataan resetoititoiminnolla. Ajamalla toimilaite maksimiasentoonsa voi lukea ohjelmasta minimi- ja maksimikulmat ja vastaavat jännitteet ja syöttää arvot anturitietoa käsittelevälle LIN_TRAFO FB:lle TEC132-logiikan anturitietoa käsittelevässä ohjelmassa. [20.]

6 Logiikkojen kytkennät ja ohjelmointi

6.1 Technion TEC132-kytkennät ja konfigurointi

Technionin logiikat on suunniteltu työkoneille ja vastaaville vaativissa olosuhteissa toimivien ajoneuvojen tarpeisiin, vaadittavalla suojausluokituksella IP67. TEC132 on ominaisuuksiltaan ja kytkentämahdollisuuksiltaan monipuolinen ja helposti konfiguroitavissa oleva logiikka (kuva 25). Valinnassa huomioitiin tulevaisuudessa mahdollisesti tulevat lisäkytkennät anturi- ja ohjauspuolella. [21, s. 5–6.]

Logiikka on sijoitettuna laitteen ulkopuolella runkoon, jossa se altistuu ulkoilman olosuhteille. Logiikan mukana toimitetaan tarvittavat johtosarjat valmiiksi kytkettynä pistokkeineen. Kytkentätöiden sujumiseksi kaikki johdot on erikseen selkeästi merkitty ja vastaavasti kommentoitu logiikan tulojen ja lähtöjen kanaville (kuva 26). [21, s.12–13].

Table 7 - TEC132 I/O configuration

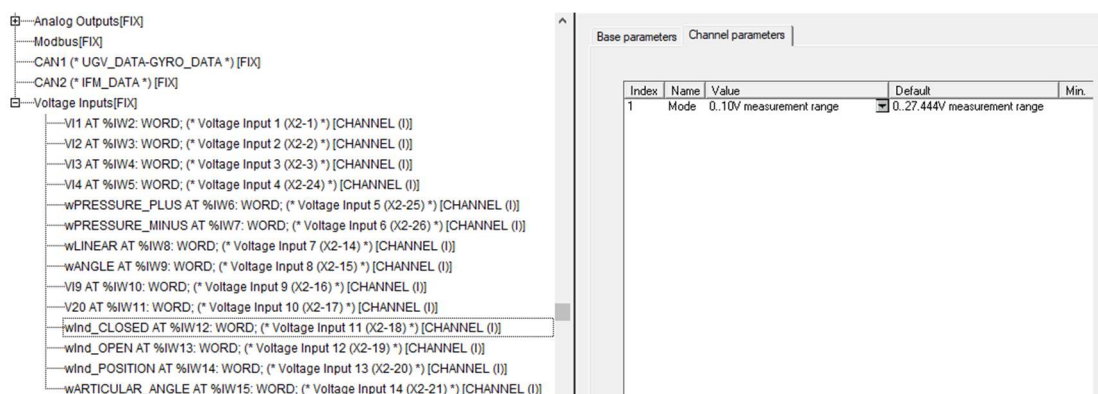
I/O Group	Pcs	I/O Types	Notes
HIGH SIDE DIGITAL OUTPUTS (Configurable input / output #1) (6 pcs)	2	DO _{H_3A5.1} DI _{H.1} AI _{10V.1} AI _{27V.1} DO _{L.1}	Digital output 3,5A, high side Digital input (PNP / active high) Analog input 0-5 V Analog input 0-27 V Digital output 1 A, low side
	4	DO _{H_3A5.1} DI _{H.1} AI _{10V.1} AI _{27V.1}	Digital output 3,5A, high side Digital input (PNP / active high) Analog input 0-5 V Analog input 0-27 V
HIGH SIDE DIGITAL OUTPUTS (Configurable input / output #2)	14	DO _{H_3A5.2} DI _{H.2}	Digital output 3,5A, high side Digital input (PNP / active high)
PRECISION INPUTS (Configurable input / #3)	8	DI _{H.3} AI _{5V.2} AI _{10V.2}	Digital input (PNP / active high) Analog input 0-5 V Analog input 0-10 V
REFERENCE OUTPUTS (Configurable output #4)	2	VREF _{5V.1} VREF _{10V.1}	Voltage reference output 5 V Voltage reference output 10 V
ANALOG OUTPUT	1	AO _{10V.1}	Analog output 0-10 V
GND	8	GND	Ground for sensors and actuators
Module power supply	2	VBAT	Power supply module logic and outputs
	2	GND	VBATT ground
CAN	2	CAN	
RS-232	1	RS232	CODESYS programming interface
RS-485	1	RS485	In selected variants only

Kuva 25. TEC132 monipuoliset konfiguroitavissa olevat liitännät [21].

Koukkulaitteen anturointi on toteutettu käyttämällä 0–10 V:n ja 2–10 V:n analogiasignaalia välittäviä antureita. Kaikilla antureilla on 24 V:n reguloitu syöttöjännite,

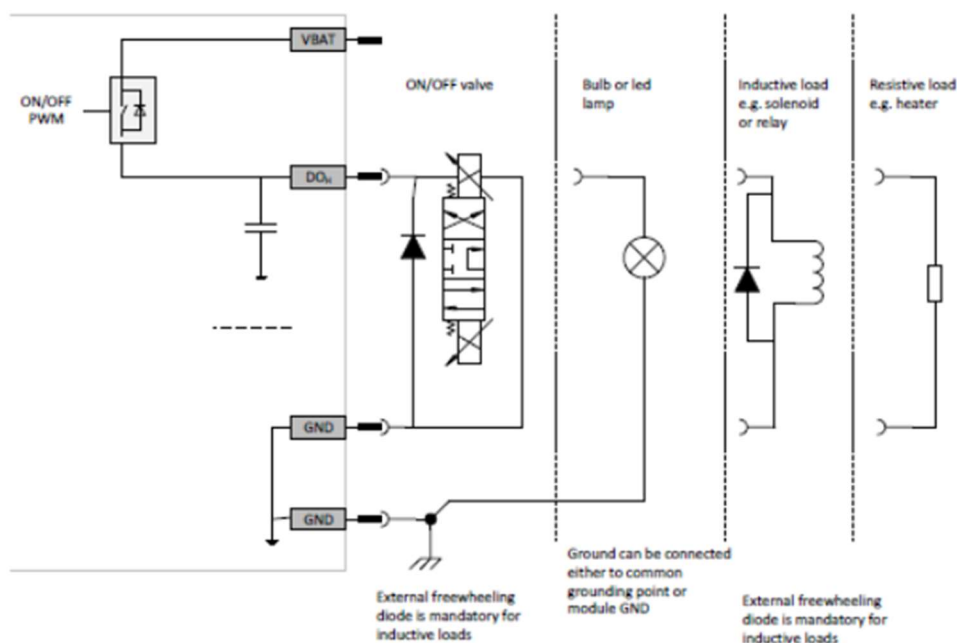
joka on kytketty logiikan DO_H lähdöiltä. Käyttämällä logiikan omaa reguloitua jännitettä, vältetään koneen jännitevaihtelun mahdollisia vaikutuksia mittausarvoihin. [21, s. 21.]

Kytkemällä antureiden syöttöjännite logiikan jännitelähtöjä hyödyntämällä, on mahdollista myös kytkeä anturit tarvittaessa pois päältä ohjelmallisesti esimerkiksi huolto- tai säätötöiden yhteydessä, ilman koko järjestelmän sammuttamista. Kanavia valittaessa on tärkeää tukeutua logiikan ohjemanuaaliin, jossa eri sisääntulojen (*Input pins*) ja lähtöjen (*Output pins*) erilaiset konfigurointimahdollisuudet ovat nähtävissä [21, s. 32.] Kanavat on kommentoitu suluissa niitä vastaavilla johdinmerkinnöillä. Jos kanava on konfiguroitavissa sekä lähdöksi että sisääntuloksi, olisi suositeltavaa nimetä tai muulla tavalla merkitä kytkentä myös käyttämättömälle optiolle, jottei myöhemmässä vaiheessa esimerkiksi lisäkytkentöjä tehtäessä lue vapaaksi merkitsemätöntä lähtöä, joka on todellisuudessa käytössä anturin sisääntulosignaalille. Esimerkki konfiguroidusta jännitesisääntulosta (Kuva 26).



Kuva 26. Technion TEC 132 PLC kanavien konfigurointi ja parametrisointi.

Koukkulaitteen hydraulikkaventtiilien ohjauskelat ohjattiin DO_H (*Digital Output High*) lähdöillä. Induktiivisessa kuormassa on huomioitu syntyvät jännitepiikit virran katketessa [21, s. 22.] Tähän tarkoitukseen on asennettu maan ja positiivisen navan välille suojadiodi, jolla suojataan logiikan lähtöä rikkoontumasta (Kuva 27). Alustan hydraulikkapiiriin kytkettiin yksi lähtö PWM-ohjauksella.



Kuva 27. Kytentäesimerkkejä erilaisille kuormille [21].

Kaikki anturit ja toimilaitteiden maadoitukset on kytketty näille varatuille I/O Ground-pinneille. Tämä suojaa vastaavalla tavalla kuin jännitteen regulointi, etenkin anturitietojen tarkkuutta. [21, s. 26.]

Kommunikointiin logiikkojen välillä käytettiin kahta CAN-väylää, CAN 1 on kytketty IFM CR1083-näyttömoduulille, josta luetaan alustan ja kohteen välistä etäisyyttä ja poikkeamaa sivusuunnassa. Toinen väylä CAN 2 on kytketty alustan Exertus-logiikalle ja gyro: lle. Exertus-logiikalta luetaan käyttäjän antamia komentoja lastauksen tai purkamisen sekvenssien aloittamiseksi ja kauko-ohjaimella tapahtuvan manuaalisen ajon komentoja. Exertus-logiikalle lähetetään tiedot lavakäsittelyn statuksesta. Gyrolta luetaan alustan pituussuuntaista kallistustietoa.

6.2 Ohjelmointi

Ohjelmointiympäristönä käytettiin Codesys 2.3: a. Ohjelmointi on toteutettu ST- ja FBD-ohjelmointikielillä. Technionin logiikat toimitetaan laitekohtaisilla kohdepaketeilla (*Target Packages*), jotka sisältävät kyseisen logiikkaan tarvittavat konfigurointitiedostot ja yleisiä kirjastoja. TEC132-logiikan ohjelma suorittaa joko kauko-ohjaimelta pyydytyt

manuaaliset toiminnot alustan Exertus PLC:n kautta, tai CAN-väylältä tulevan viestin perusteella koukkulaitteelle pyydettyvät automaattiset nosto- tai laskusekvenssit.

Ohjelma prosessoi logiikalle kytkettyjen antureiden välittämiä 0-10 V:n analogiasignaaleja. Koukkulaitteelta luetaan asematietoja liukupalkin asennosta ja kipin kallistuksesta sekä induktiivisten antureiden läsnäolon tunnistusta ja alustan rungon kääntönivelelle asennetun kulma-anturin asematietoa. Anturitietojen pohjalta suoritetaan toimilaitteiden hydraulikkaventtiileiden ja alustan PWM venttiilin ohjaukset. Lisäksi kommunikoidaan CAN-väylän yli alustan asentotietoa kytketystä gyrolta ja laitteen asemaa havaittujen objektien suhteen IFM CR 1083 näyttömoduulilta.

6.3 Alustan paikoitus nostoasemalle

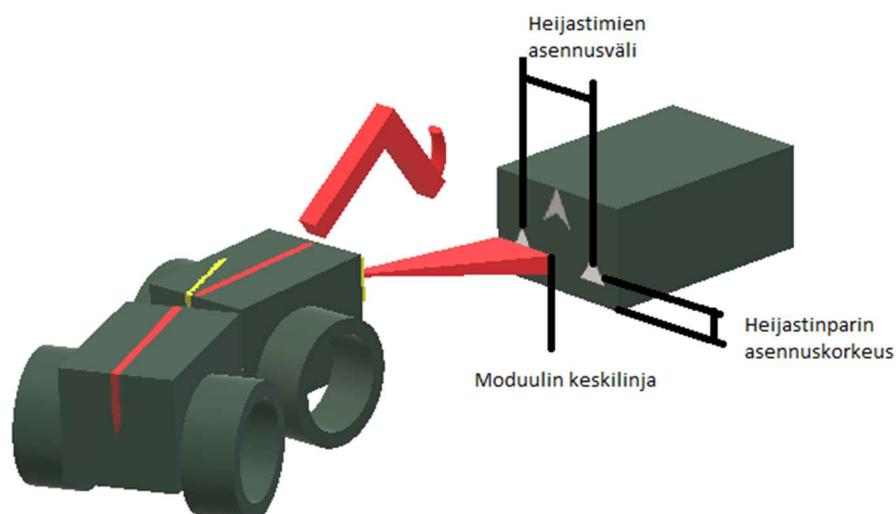
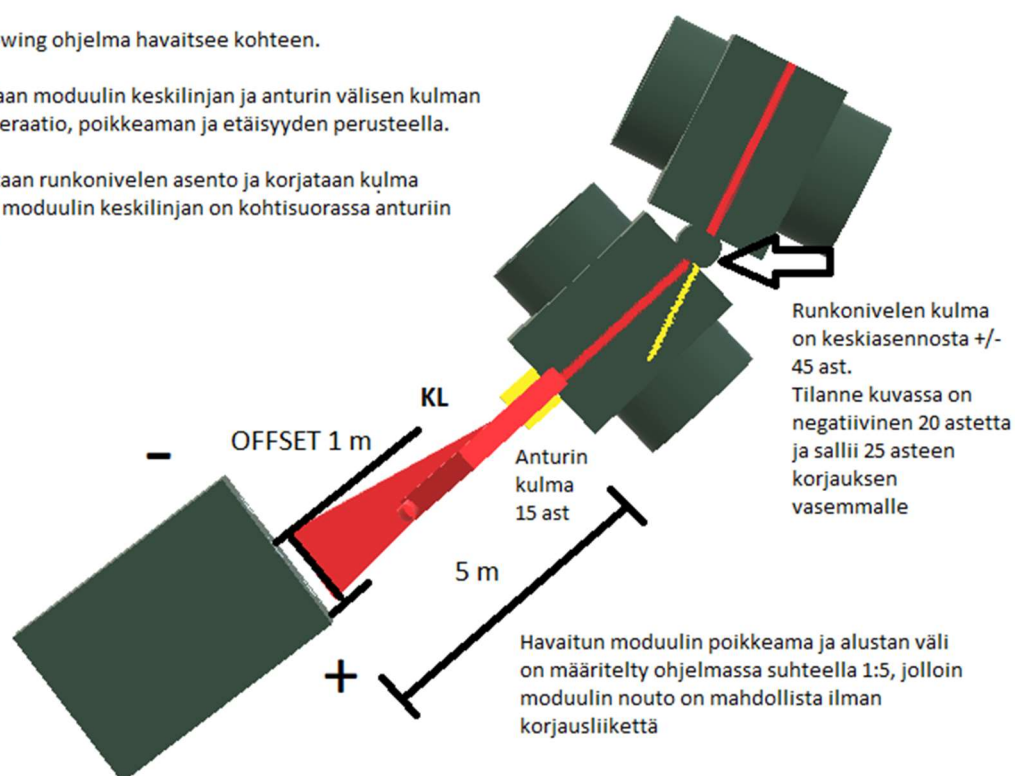
Proseduuri ajoneuvon navigoimiseksi nostoasemalle tapahtuu seuraavasti:

- Operaattorin saatuaan tiedon 3D-anturin löytäneen noudettavan kohteen, annetaan kauko-ohjaimen välityksellä komento aloittaa kohteen lähestyminen ja suorittaa lastausoperaatio.
- Anturitietojen perusteella suoritetaan matemaattiset operaatiot kohteen ja alustan välisestä asemasta, ja jos ehdot täyttyvät, paikoitus aloitetaan (kuva 28).

AgvFollowing ohjelma havaitsee kohteen.

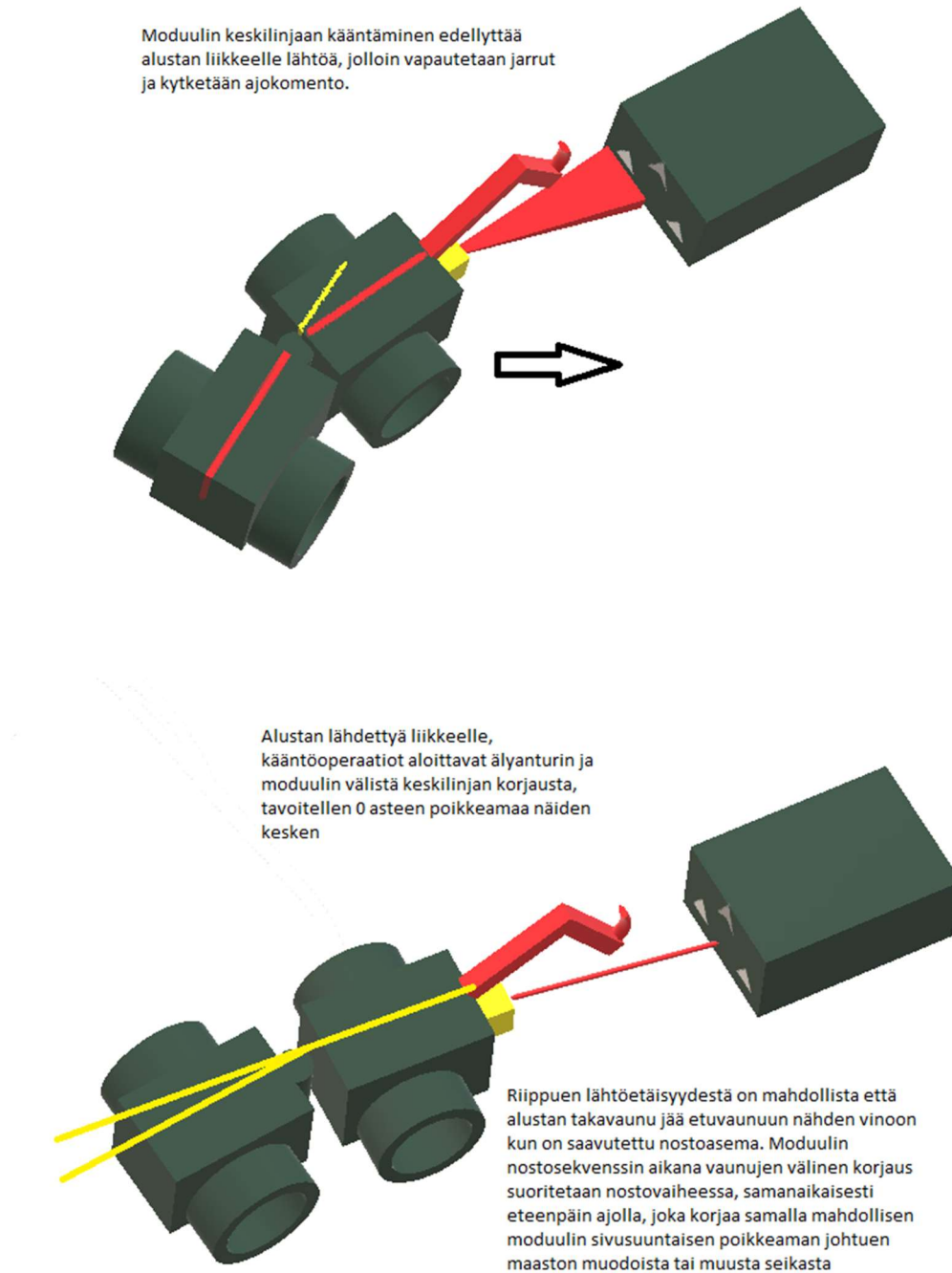
Suoritetaan moduulin keskilinjan ja anturin välisen kulman lasku-operaatio, poikkeaman ja etäisyyden perusteella.

Tarkistetaan runkonivelen asento ja korjataan kulma niin että moduulin keskilinjan on kohtisuorassa anturiin nähden.



Kuva 28. Havainnekuva 1.

- Alustan jarrut vapautetaan ja kytketään ajopumppu peruutukselle vakio PWM-arvolla, jolloin runkonivelen ohjaussylinterit kytkeytyvät ohjelman ehtojen mukaan ohjaten alustaa kohti moduulin keskilinjaa (kuva 29).

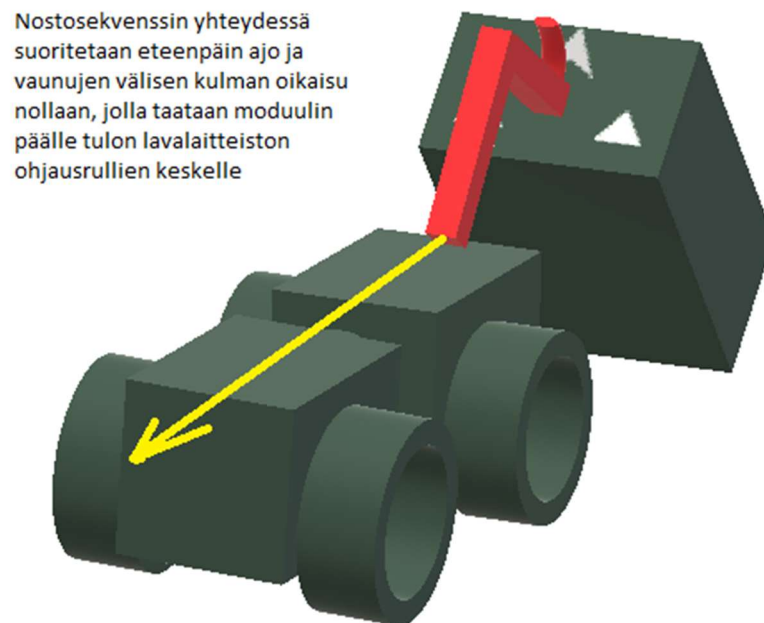


Kuva 29. Havainnekuva 2.

6.4 Nostosekvenssi

Nostosekvenssi toimii seuraavasti:

- TEC132 lukee operaattorin antaman CAN-viestin, joka antaa käskyn aloittaa alustan paikoitus ja nosto, kun asema on saavutettu.
- Tarkistetaan että alustalla ei ole ennestään kuormaa, induktiivisen anturin välityksellä juoksuilla olevasta kuormasta.
- Avataan takalukitus, jos se on kiinni. Lukituksella on sekä auki- että kiinniasennoille induktiiviset anturit.
- Ajetaan liukupalkki taka-asentoon, jos se on ulkona. Asento mitataan lineaarisella vaijerianturilla.
- Ajetaan kippisylinteri taakse ja alas nostokorkeudelle. Nostokorkeus tarkistetaan vertaamalla alustan gyrotietoa ja kippirungon välistä kulmaa.
- Liukupalkki ajetaan ulos lavan ja laitteen välisen mitatun tiedon perusteella, joka lähetetään kameralta CR1083:n kautta CAN-viestinä
- Aloitetaan kippirungon nosto 20 cm, pysäytetään ja mitataan kippisylinterissä valitseva paine, joka on korkeampi kuin tyhjällä puomilla, jos koukku on tarttunut lavan tartuntaan.
- Kun on todettu, että puomi on saanut lavan tartunnasta kiinni, tarkistetaan alustan rungon keskinivelen asento ja ajetaan se tarvittaessa suoraksi. Aloitetaan koneen eteenpäin ryömintä (kuva 30) ja jatketaan nostoa, samalla liukupalkki ajetaan takaisin sisäasentoon. Tällä toimenpiteellä varmistetaan, että päälle vedettävä lava on alustan kanssa linjassa takarullien päälle noustessa.
- Koukkupuomi ajetaan alas ja tarkistetaan lavan asentoa valvovalla induktiivisella anturilla, että lavan juoksu on paikallaan kannakkeen päällä.
- Liukupalkki ajetaan sille määrätyle mitalle, joka vaihtelee moduulin pituuden mukaan.
- Lukitaan takalukot.
- Annetaan alustan päälogiikalle tieto CAN viestinä, että siirto on mahdollinen.



Kuva 30. Havainnekuva 3.

6.5 Laskusekvenssi

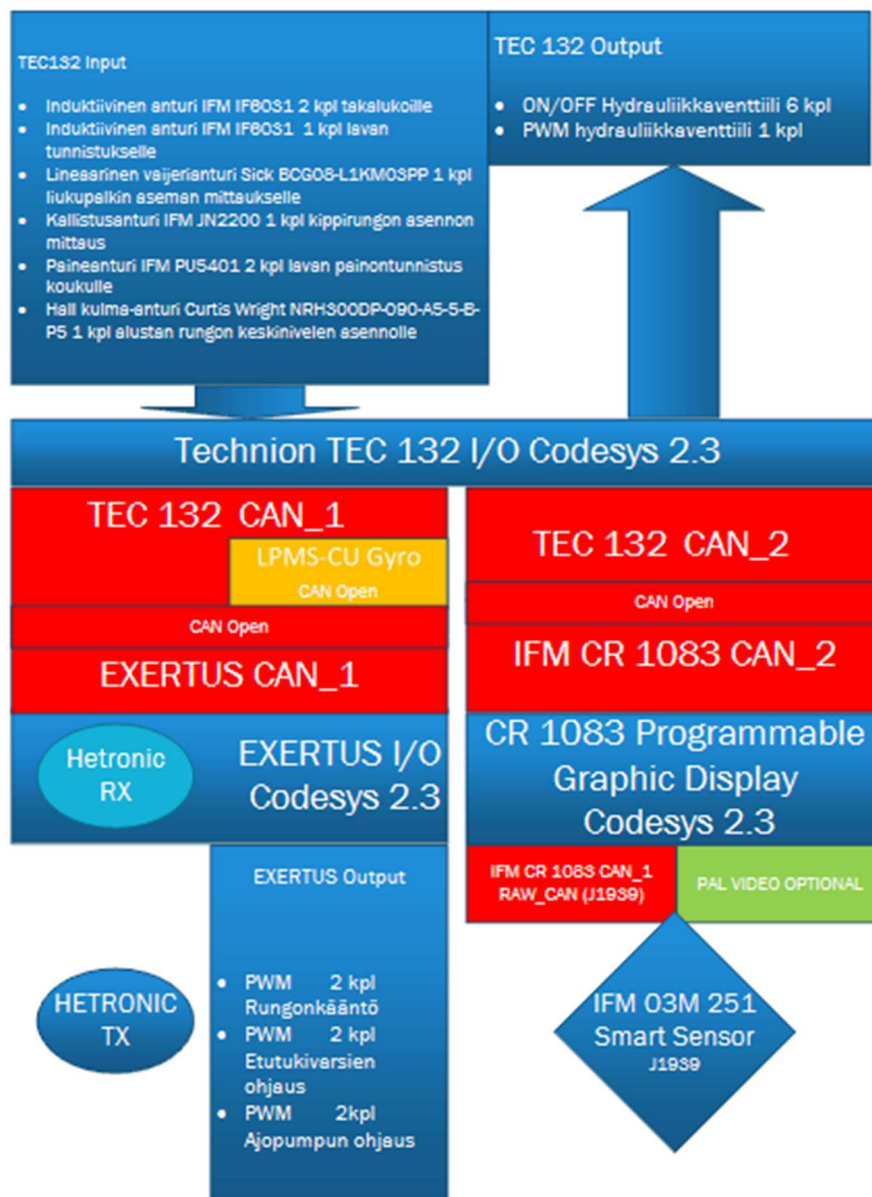
Laskusekvenssi toimii seuraavasti:

- TEC132 lukee operaattorin antaman komennon CAN väylältä, että kuorman voi laskea.
- Avataan takalukot.
- Ajetaan liukupalkki taka-asentoon.
- Ajetaan kippirunkoa alas ja pysäytetään 2 cm ennen kuin lavan etureuna on maassa ja ajetaan liukupalkkia ulos 25 cm.

- Ajetaan kippirunko alas, tunnistetaan paineen putoaminen koukun irrotessa lavan tartunnasta.
- Ajetaan liukupalkki sisäasentoon irti tartunnasta.
- Ajetaan kippirunko eteen ja alas.
- Annetaan alustalle lupa siirtyä.

7 IFM CR1083-näyttömoduulin kytkennät ja konfigurointi

Kamerajärjestelmän datan prosessointiin ja välittämiseksi valittiin IFM CR1083-näyttömoduuli, jossa on 4 kpl CAN-liitäntää, Ethernet, USB, video ja USB-muistilaitteelle kytkennät. Moduuliin on kytketty 2 kpl CAN-väylää, joista CAN1-liitäntä toimii samalla sekä CAN-signaalien että virransyöttönä O3M-anturille. CAN2 välittää datan TEC132:lle. [22, s. 14.] Kaapeloinnissa on huomioitu tarvittava 60 Ω :n kokonaisvastus väylällä. Logiikoiden, älyanturin ja muiden laitteistojen fyysiset kytkennät ja arkkitehtuuri keskenään on havainnollistettuna kuvassa 31.



Kuva 31. Logiikkojen ja komponenttien kytkentä ja arkkitehtuuri.

7.1 Ohjelmointi ja apuohjelmat

Object Detection Firmware paketin demoprojekteista hyödynnettiin ifm_CR1085_O3M_Featurepack_ObjectList-apuohjelmaa. Ohjelman sisälle integroitiin toinen, IFM:n kehittämä AGV Follow Sample Project Version 1.1, tehtävänä tietyllä mitalla asennettujen heijastimien paikantaminen ja koneen asematiedon laskeminen näiden suhteen.

Heijastimien havaitseminen heikkenee olennaisesti alle 2 m:n matkalla [15, s. 2]. Alustan ohjaus suoritettiin loppuun ifm_CR1085_O3M_Featurepack_ObjectList-apuohjelmalla. Apuohjelmassa prosessoidaan normaalia objektin tunnistusta, lavan etäisyyden mittaukseen viimeisen 1,5 m:n pituudelta. Mittausalueella olevien muiden normaaliin objektien pois rajaamiseksi, ohjelmaan on luotu lavan etuseinän tunnistamiseksi mitta ja ID-ehdot. Anturin on havaittava mittausalueelta tietyn levyinen ja etäisyydellä olevan objekti tietyllä tunnistenumeroilla (ID) (kuva 32).

```
IF ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_x1_cm <= 150 AND
ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].by_id = 72 AND
ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_x2_cm >=
((ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_x1_cm + 8) AND
((ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_y1_cm * -1) +
ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_y2_cm) >= 104 THEN

i_WIDTH_OBJ_72 := ((ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_y1_cm * -1) +
ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_y2_cm);

x_MODULE_TRUE := TRUE;

.AGV_20;
AGV_FBD_20;

ELSIF ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_x1_cm > 150 THEN

x_MODULE_TRUE := FALSE;

PLC_PRG_AGV_Follow();
prgAgvFollowing();
AGV_10;
AGV_FBD();

END_IF
```

Kuva 32. Alustalle luodut ehdot jatkavat moduulin lähestymistä ilman Follow me ohjelman opastusta.

Object List-ohjelmaa on hyödynnetty myös CR1083-näytön välityksellä antamaan käyttäjälle objektien paikkatietoa. Kahdeksan objektin taulukossa tulostetaan ensin löytyneet heijastimet tai heijastimia vastaavia havaintoja. Kynnysarvolla joka määritetään anturia konfiguroitaessa, suoritetaan pintojen heijastavuuden analysointi itse anturissa. Kyseisen Object List-ohjelman on mahdollista havaita heijastimet ennen Agv Following-ohjelmaa, johtuen Agv Following-lohkoon syötetystä maksimi kaistan leveydestä, josta sen pyydetään tekevän havaintoja. Projektissa käytettiin 5 m:n kaistaa, joka puolitettiin anturin keskiliinjalta oikealle ja vasemmalle.

Objektin tyyppi lähetetään viestikehyksessä bittinä 0x0 = regular object tai 0x1 = retroreflector (kuva 33).

Bit 7	Obj_*_Type	enum	P = C	Type identifier. 0x0 = regular object 0x1 = retroreflector
-------	------------	------	-------	--

Kuva 33. Objektin luokittelu joko normaaliksi tai heijastavaksi [17].

Näyttöön välitetystä datasta on huomattava apu maastossa tapahtuvissa testeissä, se mahdollistaa objektien havaintoja ilman laitteeseen kytkettyä tietokonetta.

AGV Follow-muuttujilta luetaan asematietoja, joita hyödynnetään omissa apuohjelmissa ohjaamaan alustaa halutun moduulin eteen noutoalueelle. Tietoja luetaan havaitusta heijastinparista (AGV_Detected), etäisyydestä (AGV_Distance) ja keskilinjän poikkeamasta anturiin nähden (AGV_Displacement). Alustalla ei ole nopeuteen perustuvaa anturointia. Lähestymisessä AGV_Distance-muuttuja on AGV Follow-ohjelman valmiiksi laskettu etäisyyden asema.

Object List-ohjelman muuttujalta Object [0] hyödynnetään kohteen ja alustan välisen etäisyyden mittaamiseksi ja tunnistamiseksi lavan etuseinän syvyysuuntaista tietoa, leveystietoa ja objektin ID-numeroa (kuvat 34 ja 35).

```

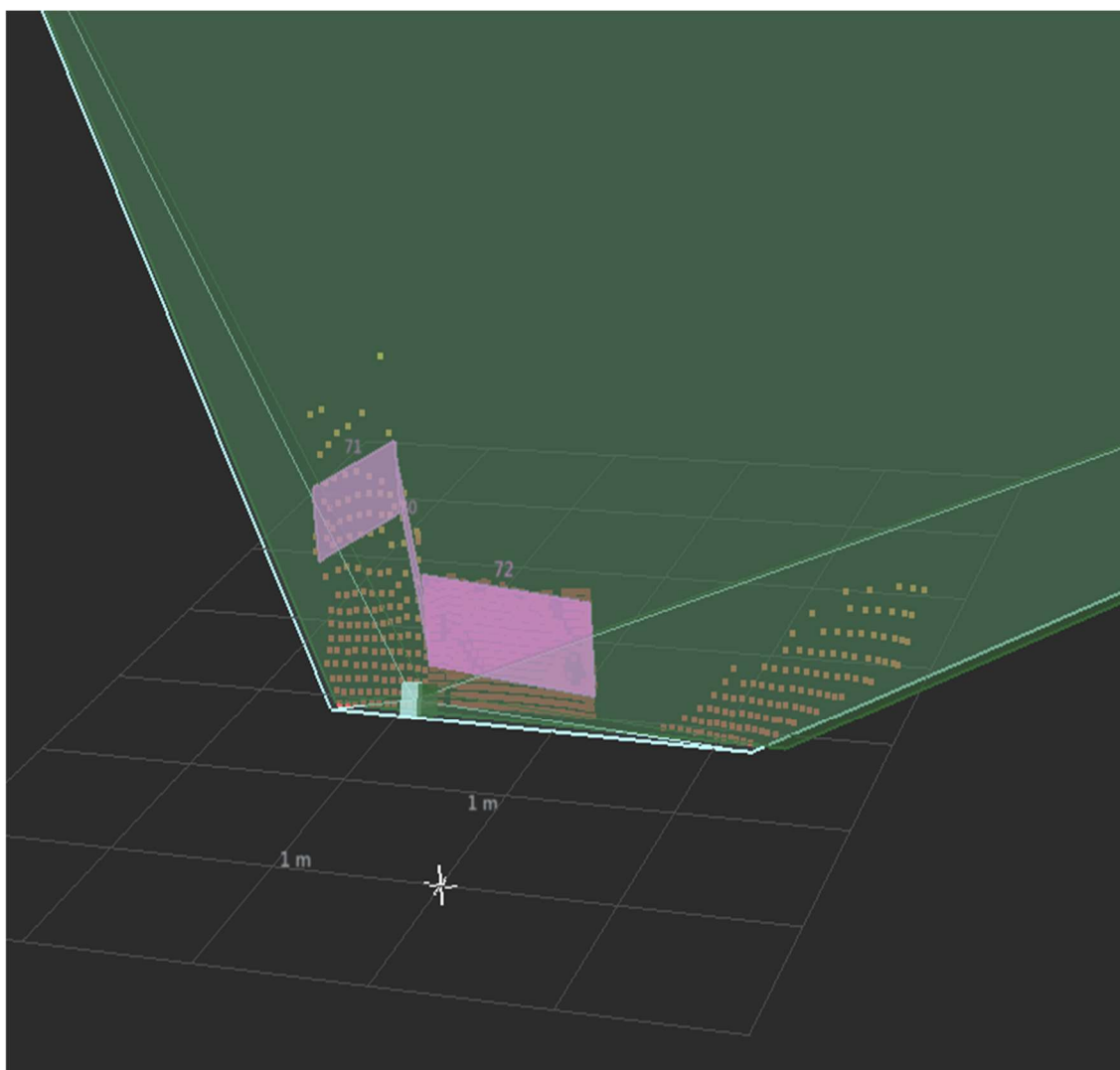
-o_fb_CANReadObjectInfo
  -i_by_canChannel = 1
  -i_by_numObjects = 8
  -i_st_msgIDs
    -i_e_MinEP = ep_0to25
    -i_e_MinTrackAge = age_0to2_frames
    -o_ast_o3mObject
      -o_ast_o3mObject[0]
        -bo_error = FALSE
        -by_id = 72
        -e_type = objectType_normal
        -bo_objectToggle = FALSE
        -e_trackAge = age_greater25_frames
        -e_existanceProbability = ep_95to100
        -bo_measured = TRUE
        -i_x1_cm = 138
        -i_y1_cm = -46
        -i_x2_cm = 148
        -i_y2_cm = 58
        -i_zMin_cm = 40
        -i_zMax_cm = 88
        -i_vx_cmps = 0
        -i_vy_cmps = 0
        -i_vz_cmps = 0
        -e_vxQuality = vq_95to100
        -e_vyQuality = vq_95to100
        -i_ax_cmps2 = 0
        -i_ay_cmps2 = 0
        -i_az_cmps2 = 0

```

Kuva 34. Object List-ohjelman ensimmäisen havaitun objektin (Object [0] :n) valmiiksi laskettu data.

Lopullisen aseman virhe X-suunnassa (alustan ja moduulin kohtisuora etäisyys toisistaan), korjataan nostosekvenssin ohjelmassa ajamalla koukkuliukupalkkia nostettavan moduulin todellisen aseman ja alustan halutun aseman virheen matkan ± 1 cm moduuliin kiinnittyessä. Virhemarginaali on laskettu 10 asemointikerran keskiarvona.

Mittausvirhe syntyy 3D-anturin epätarkkuudesta lähietäisyyksillä ja moduulin todellisen aseman virhe alustan nopeusvalvonnan puuttumisen takia, jolloin joudutaan turvautumaan 3D-anturin antamaan etäisyystietoon lähestymisen aikana ennen pysähtymistä.



Kuva 35. Kuva 1 Object [0] ID 72.

7.2 Järjestelmän eri CAN-protokollat

7.2.1 CAN Open

Kaikki logiikat ja alustan gyro ovat keskenään yhteydessä CAN Open-protokollalla. Tämä mahdollistaa eri valmistajien ja komponenttien tiedon välittämisen ja tarvittavien viestien vastaanottamisen. Technionin TEC132-logiikassa ei ole mahdollisuutta EDS-tiedostojen käsittelyyn. TEC132:lta lähteville CAN-viesteille on luotuna CAN Enable ja TX-funktioblokeilla rajapinnat, käyttäen FBD-ohjelmointikieltä, ja RX-rajapinnat on luotu perinteisellä ST-ohjelmointikielellä. Lähtevät ja luettavat CAN-viestit ovat 8-Byte:n pituisia.

7.2.2 Raw CAN ja J1939

Raw CAN on protokolla, jossa poimitaan J1939 CAN-viestin 11–29 kehykset. Tässä kokoonpanossa O3M-anturi on oletuksena valmiiksi konfiguroitu J1939 protokollalla, ja tarvittavat kirjastot CR 1083-näyttömoduulin apuohjelmassa mukana viestien tulkintaan.

O3M-anturi lähettää prosessoidun datan J1939 viestikehyksinä. Visual Assistantilla anturin konfigurointi ja muut parametrisoinnit käyttävät myös samaa protokollaa Ethernetin rinnalla.

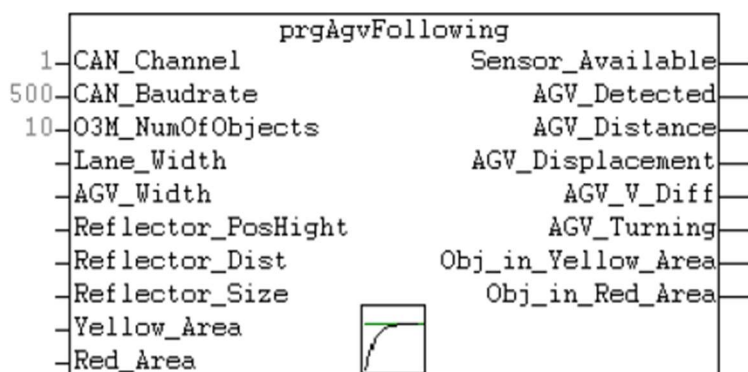
8 Ohjelmien arkkitehtuuri ja suoritus

8.1 CR1083 PLC ja O3M 251-anturi

Prosessoitu anturitieto luetaan CR 1083-logiikalla, J1939-viesteistä Raw CAN-protokollalla. IFM-apuohjelmat hyödynnetään tarvittavan anturitiedon poimimiseksi omalle ohjelmalle. Logiikka toimii porttina, jossa laskettu etäisyys kohteeseen, sivuttaispoikkeama kohteen keskilinjasta ja ajon aikana tapahtuvan keskilinjalta poikkeamisen lasketut arvot välitetään Technion TEC132-logiikalle CAN-viestinä USINT-muodossa.

Ohjelmassa suoritetaan etäisyyteen ja poikkeamaan perustuva laskuoperaatio. Kyseinen operaatio karsii pois paikoitus yrityksen, kun alustan tiedetään sijaitsevan asemassa, josta ei kykene yhdellä etenemisellä paikoittamaan alustaa mahdolliselle nostoasemalle. Parametrit on kirjoitettu FBD-ohjelman ehtoihin ja perustuvat testeissä toimiviksi todettuihin arvoihin. Parametreihin vaikuttaa laitteen kääntösäde, pituus, nopeus ja anturin kyky pysyä lukittuna käännön aikana kohteessa.

AGV Following-ohjelma suorittaa laskemat havaitun heijastusparin keskilinjan poikkeama ja etäisyys. Ohjelmapaketissa on valmis FB, jolle syötetään tässä projektissa heijastimien asennusmitat, tutkittavan alueen leveys ja alustan leveys (kuva 36). Halutessa voi vielä määritellä etäisyyteen perustuvia alueita kohteelle. Alkuperäinen funktio näille alueille on laitteen nopeuden määrittäminen eri etäisyyksillä edessä ajavan ajoneuvon suhteen. [23.]



Kuva 36. IFM:n kehittämä Sample project "AGV following" V1.1 FB [23].

Lohkolla luetaan ulostuloilta valmiiksi lasketut arvot (Kuvat 37 ja 38). Tässä projektissa käytetään AGV_Detected-ulostuloa välittämään tiedon, että kohde, joka täyttää annetut kriteerit on havaittu ja tieto lähetetään yhdellä bitillä. Etäisyys- ja poikkeama-arvot poimitaan aliohjelmalle AGV_1, jossa suoritetaan muunnokset ja välitetään USINT-muodossa, neljällä eri muuttujalla DISTANCE_CM, DISTANCE_M, DISPLACE_CM ja DISPLACE_M koukkulaitteen TEC132-logiikalle. Lisäohjelmointityö on liitteenä 2.

Input parameters:

Name	Type	Description
CAN_Channel	BYTE	CAN channel with O3M connection
CAN_Baudrate	WORD	Baudrate of CAN channel
O3M_NumOfObjects	BYTE	number of objects of O3M
Lane_Width	REAL	width of driving lane [m]
AGV_Width	REAL	width of AGV [m]
Reflector_PosHight	REAL	height of reflector position [m]
Reflector_Dist	REAL	distance between reflectors [m]
Reflector_Size	REAL	size of reflector [m]
Yellow_Area	REAL	yellow area (slow down) [m]
Red_Area	REAL	red area (stop) [m]

Kuva 37. Agv following ohjelmalle tarvittavat parametrit [23].

Lasku tapahtuu muuttamalla REAL arvo metreistä kertoimella 100, senttimetreiksi. Jako suoritetaan modulo-jakolaskulla (*Modulo operation*) MOD 100, jonka jäännösarvo on cm ja tulos m. Väylän yli siirrettävää dataa halutaan rajata yhteen 8 tavun viestiin. Suorittamalla osan laskuprosesseista CR 1083 moduulilla, pystytään toimittamaan pienempiä yhteen tavuun (Byte) sopivia arvoja purettavaksi TEC 132:lle. Laitteen paikoittamiseen tarvittava mittausdata riittää senttimetrin tarkkuudella, joten on edullisempaa tehdä muunnos väylän liikenteen säästämiseksi.

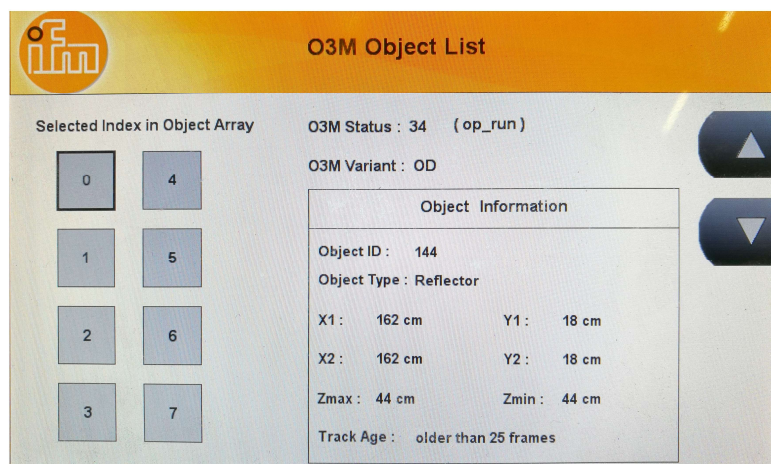
Output parameters:

Name	Type	Description
Sensor_Available	BOOL	sensor is available
AGV_Detected	BOOL	AGV in front detected
AGV_Distance	REAL	distance to AGV in front [m]
AGV_Displacement	REAL	displacement of AGV in Front to the side [m]
AGV_V_Diff	REAL	relative velocity to AGV in front [m/s]
AGV_Turning	INT	AGV in front is turning [0=no; 1=left; 2=right]
Obj_in_Yellow_Area	BOOL	object in yellow area detected
Obj_in_Red_Area	BOOL	object in red area detected

Kuva 38. Agv Following:n luettavat ulostulot [23].

Pääohjelman sisällä luetaan omia aliohjelmia AGV_SELECTION-ohjelman ehtojen mukaisesti. AGV-ohjelmat suoritetaan, kun anturi on havainnut kohteen, johon lukittua, ja alustan lähestyminen aloitetaan. AGV_2 ja AGV_FBD_2 ovat apuohjelmia alustan ja

noudettavan kohteen välin pienentyessä 1,5 m:iin. Ominaisuutena 3D-anturi ei enää erota lähietäisyydellä heijastinta verrattuna tavallisiin pintoihin. Testeissä osoittautui käytetyllä heijastimien asennusmitoilla havainnon loppuvan noin 1,4 m:n paikkeilla. Apuohjelman suoriutuessa ei enää ohjata alustaa kuin etenemissuunnassa. Tieto luetaan normaalilta objektien ensimmäiseltä ruudulta, joka vastaa lavan etuseinää ja sen etäisyyttä. Apuohjelmalle on luotuna näyttömoduulille käyttöliittymä, josta voidaan tarkastella kohteiden tyypit ja koordinaatit (Kuva 39).



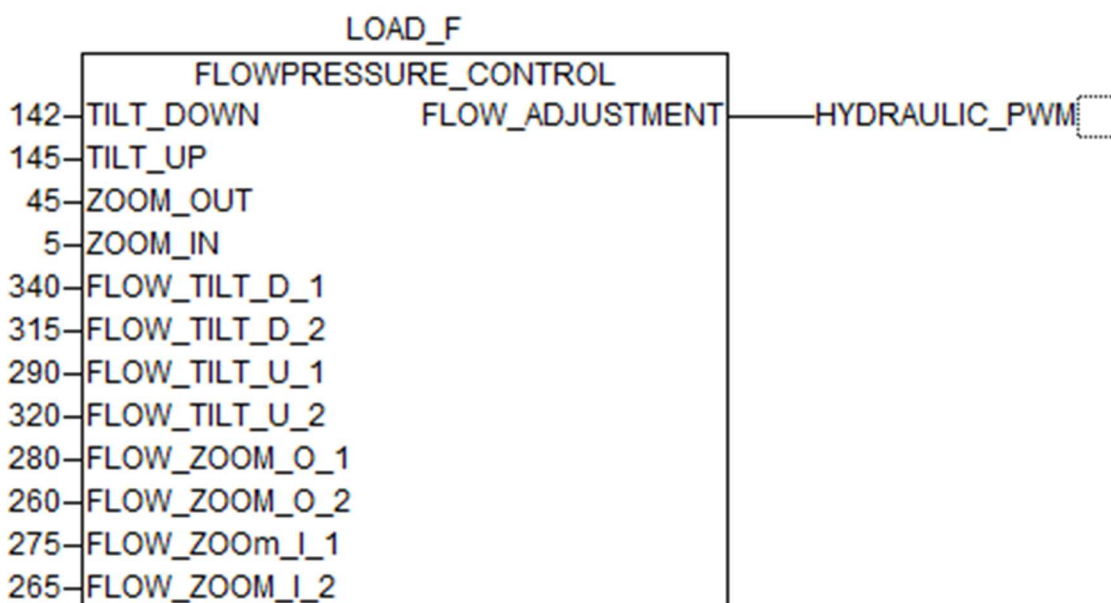
Kuva 39. IFM CR1083 näyttömoduulin hyödyntäminen kenttätesteissä objektien paikantamisessa.

8.2 Technion TEC 132

Technionin logiikka suorittaa järjestelmän pääohjelmia, joissa prosessoidaan muiden logiikoiden ja antureiden välitetyt viestit. Ohjelman koodi on tallennettu liitteeseen 3. Pääohjelma lukee ja suorittaa kahdeksaa aliohjelmaa ja näiden sisälle luotuja aliohjelmia, riippuen käyttäjän pyytämästä moodista, joita on kolme: lastaus (load), purku (unload) ja käsikäyttö (manual).

Koukkulaitteen hydraulikkasyylintereiden nopeudensäätöön eri moodeilla on luotu FB, jolla voidaan määrittää haluttu virtauksensäätö hydraulikkaventtiilin PWM-arvoon (kuva 40). Sekä kipin kallistuskulman että liukupalkin pituusmittauksen perusteella voidaan myös määrittää, missä asemassa halutaan lisätä tai vähentää PWM-arvoa, liikkeiden nopeuttamiseksi tai hidastamiseksi.

Funktiolohko kutsutaan kuhunkin ohjelmaan erikseen, jolloin liikkeiden nopeus voidaan määrittää kuormalla ja tyhjällä puomilla, eri sekvenssien ja manuaaliliikkeiden aikana. Liikkeiden loppuun on säädetty pienempi virtaus, jolla saadaan pehmeämpi lopetus ja vähemmän rasituksia mekaanisille osille.



Kuva 40. PWM-arvoille luotu Flow_Adjustment FB, jolla säädetään hydrauliiikan liikenopeuksia annettujen parametrien mukaisesti.

Alustan runko-ohjaukselle on luotuna aliohjelma, jossa hydrauliiikkasyntereitä ohjaavien venttiilien aktivoituminen suoritetaan päälle ja pois kytkettävällä PWM-arvolla. Koska alustalla ei ole luotettavaa tapaa mitata nopeutta ennen kuin on riittävän lähellä kohdetta, perinteistä matematiikkaa ei voida soveltaa laitteen ohjaamiseksi. Ohjelman ohjaus ehdot perustuvat 3D-anturin antamaan tietoon kohteen poikkeamasta alustaan nähden, jotka luetaan väylältä USINT-muodossa CR 1083:n välittämänä.

Ohjelmassa suoritetaan tangentin laskuoperaatio, jonka tulos jaetaan ATAN-blokilla (kaava 1) radiaaneiksi ja konversio kertoimella 57.295779 ($180^\circ/\pi$), muuttaen arvon asteiksi. Tuloksen arvo kertoo alustan keskinivelen aseman asteissa, heijastinparin keskilinjan suhteen. Arvoa käytetään ohjelmalohkossa, jossa verrataan alustan rungon asemaa (älyanturin kulma suhteessa moduulin keskilinjaan) ja tavoitettavan kohteen asemaa, suorittaen annettujen ehtojen mukaisesti kääntöoperaatioita kohteen keskilinjan saavuttamiseksi.

Poikkeama-arvot ohjelmassa ovat keskilinjasta vasemmalle positiivisia ja oikealle negatiivisia. Negatiiviset luvut luetaan TEC132:lla positiivisina USINT-arvoina ja kerrotaan -1:llä kun bitti muuttujalla `i_NEG_DISPLACEMENT` on tosi. Muuttuja välitetään IFM-ohjelmasta, jossa tarkistetaan positiivinen tai negatiivinen poikkeama. Mittaustieto lähetetään USINT (0–255)-muodossa joka vastaa 0–255 cm.

ATAN

[CoDeSys IEC operator](#): Returns the arc tangent (inverse function of tangent) of a number.

IN can be type BYTE, WORD, DWORD, INT, DINT, REAL, SINT, USINT, UINT, UDINT. The result OUT is calculated in arch minutes and must be type REAL.

Example in IL:

```
LD 0.5
```

```
ATAN
```

```
ST q (* Result is 0.463648 *)
```

Example in ST:

```
q:=ATAN(0.5);
```

Example in FBD:

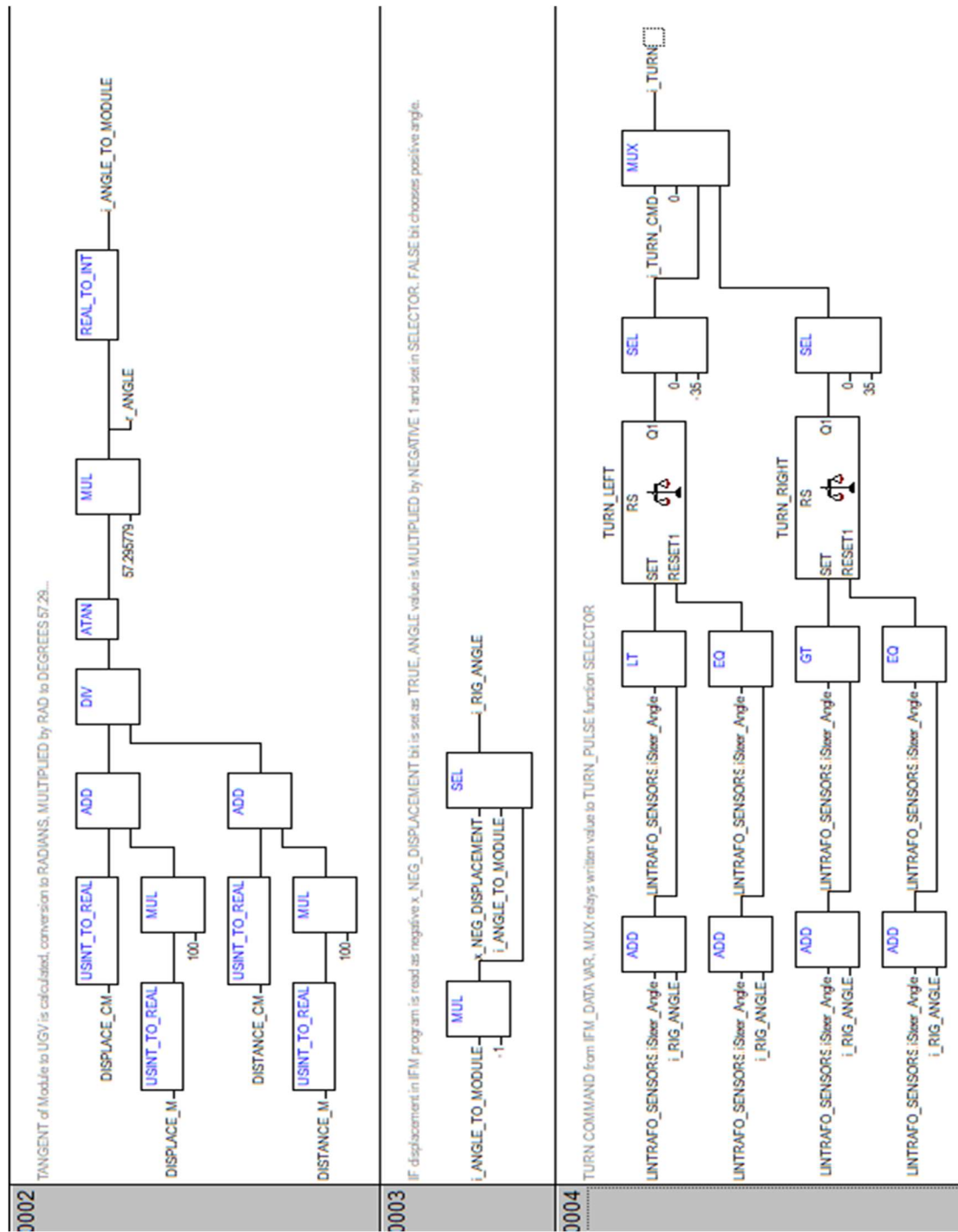


Copyright © 2010 3S-Smart Software Solutions GmbH CoDeSys V2.3.9.24 / 28-Oct-2010

Kaava 1 ATAN FB (Codesys Help).

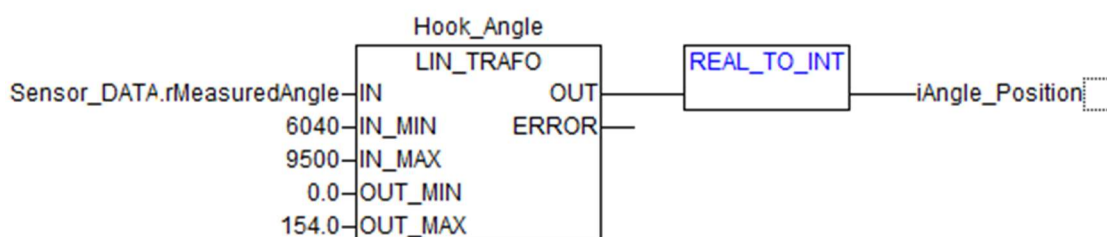
Ohjelma alustan kääntämiselle vertaa lasketun `i_RIG_ANGLE`-kulman ja alustan keskinivelen asentoa, kytkien ehtojen mukaan käännön-RS-kiikkuja, jotka vuorostaan antavat ohjauskomennon kytkeä SEL FB. SEL-lohkolla annetaan ei-aktiivisessa tilassa ulos 0 arvo ja aktiivisena PWM-ohjaukselle haluttu arvo. Molempien kääntöjen arvot valitaan vielä MUX FB-lohkolla, jonka läpi arvot ajetaan ajomoodista riippuen. Ohjelmassa `CONTROL_CONDITIONS` on luotu apumuuttuja `i_TURN_CMD`, joka ehtojen mukaan antaa arvot 0,1,2 MUX lohkolle ohjaussuunnan kytkemiseksi (Kuva 41).

Alustan lähtiessä liikkeelle joko asemointia varten tai nostotilanteessa, verrataan älyanturin asentoa lavan keskilinjan asemaan tai runkonivelen keskiasentoa. Paikoituksen tai nostosekvenssin ohjausarvo välitetään muuttujalle `i_TURN` tai `i_TURN_ALIGN`, jolloin haluttu suunta ja kyseinen ohjausventtiili kytkeytyy määritetyllä PWM arvolla. `i_TURN`-arvo ohjaa alustan kääntämistä moduulille asemoidessa ja `i_TURN_ALIGN`-arvoa käytetään alustaa eteenpäin ajettaessa, ennen lavan lopullista päälle nostoa, lavan ja alustan välisen sivuttaisen poikkeaman tai vinouden korjaamiseksi. Ajoneuvon nopeus ja kääntämiseen tarvittavat PWM-arvot asemoinnin aikana on nopeusvalvonnan puuttumisen vuoksi testattava käytännössä hyvien tuloksien saamiseksi.



Kuva 41. Alustan kääntöoperaatioita laskevat ja ohjaavat ohjelmalohkot.

Aktiivisten antureiden arvojen skaalauksessa käytetään LIN_TRAFO FB, joka laskee luetun anturin minimi ja maksimi arvot ja kääntää ne halutulle asteikolle ulostulossa, jonka halutut minimi ja maksimi arvot syötetään lohkolle (kuva 42). Arvot luetaan logiikan sisääntulolla 0–10000 WORD-muodossa ja käännetään antureille luodulla ohjelmassa REAL-arvoiksi. Arvot vastaavat 0–10 V:n jännitettä (0.0–10000.0 REAL). Esimerkkinä kippirungolle asennetun kallistusanturin asematiedon skaalauksesta (kuva 42). Lohkossa olevat IN_MIN, IN_MAX ja OUT_MIN, OUT_MAX arvot ovat anturin konfiguroinnin ja parametrisoinnin yhteydessä saatu kallistusanturin ohjelmointityökalulla.



Kuva 42. LIN_TRAFO FB kallistusanturin skaalaus.

Real-arvojen skaalauksen jälkeen muunnetaan ulostulon arvo integeriksi, jolla saadaan 1 asteen tarkkuus liikkeen asematiedosta, joka on riittävä tarkkuus ohjaukselle tässä applikaatiossa. Vastaavilla blokeilla skaalataan liukupalkin ja runkonivelen asematiedot.

Koukkulaitteen ohjaus valitaan kolmen eri moodin väliltä kauko-ohjaimelta operaattorin päätöksellä, kun saatu tieto mahdollisesta lavan havaitsemisesta tai lavan purkamisesta on saatu tai toiminnot halutaan ohjata manuaalisesti. HOOKLIFT_CONTROL toimii kolmen muun aliohjelman (HOOK_LOAD, HOOK_UNLOAD ja HOOK_MANUAL) pääohjelmana, jossa ehtojen mukaan suoritetaan kutsuttua ohjelmaa. Ohjelmat on kaikki luotu FBD-ohjelmointikielellä, ja tarvittavat muuttujat luetaan globaaleista muuttujista.

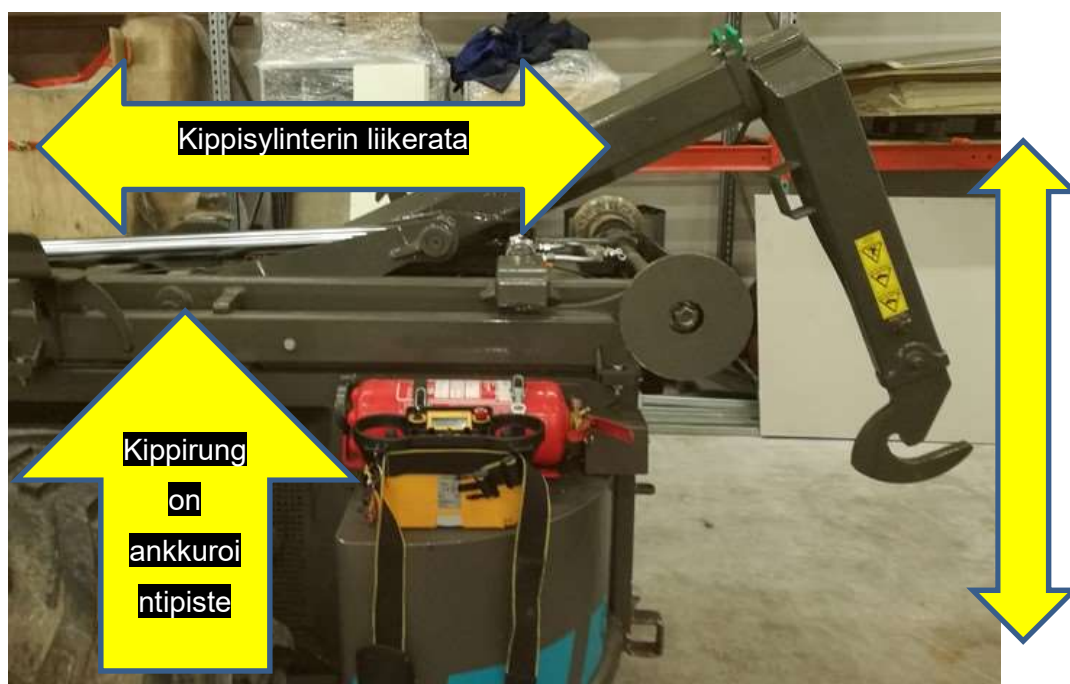
Lukituksien ohjauksille on luotu molemmille oma, perus-RS-kiikulla toimiva kytkentä, joka ohjaa haluttua venttiiliä. Ehtojen täytyessä ne kytkeytyvät pois, kun haluttu asento on saavutettu induktiivisen anturin aktivoiduttua. Jokaisella moodilla on omat ehdot lukituksien ajoille.

9 Hydrauliikka

9.1 CPS-Koukkulaite (Hook Lift)

Alkuperäinen koukkulaitteen paineistus oli 24 V:n hydrauliikkakoneikolla toteutettu, joka sai käyttövoimansa alustan akkupiiristä. Koneikon tilalle on kytketty alustan hydrauliikkapiiristä paine- ja paluuliitännät. Monoblock-venttiiliryhmän kaikki toiminnot on suojattu yhteisellä 260 bar pääpaineenrajoituspatruunalla. Laitteelta vaaditaan tarkkoja ja hallittuja liikkeitä moduulin tarttuessa ja irrotusvaiheessa. Puomiston geometria on epäedullinen juuri näissä ääriasennoissa, joissa liikkeiden tulisi olla mahdollisimman tarkkoja. Epäedullinen asento johtuu kippisylinterin ja kääntövarren välisestä suhteesta.

Kippisylinterin liikerata on kääntövarren ollessa takana ja ala-asennossa lähes kohtisuorassa linjassa kääntövarren ankkurointipisteen kanssa, mikä aiheuttaa suuren poikkeaman kääntövarren runkoon kiinnitetyn koukun pystysuuntaisessa liikkeessä pienilläkin muutoksilla.



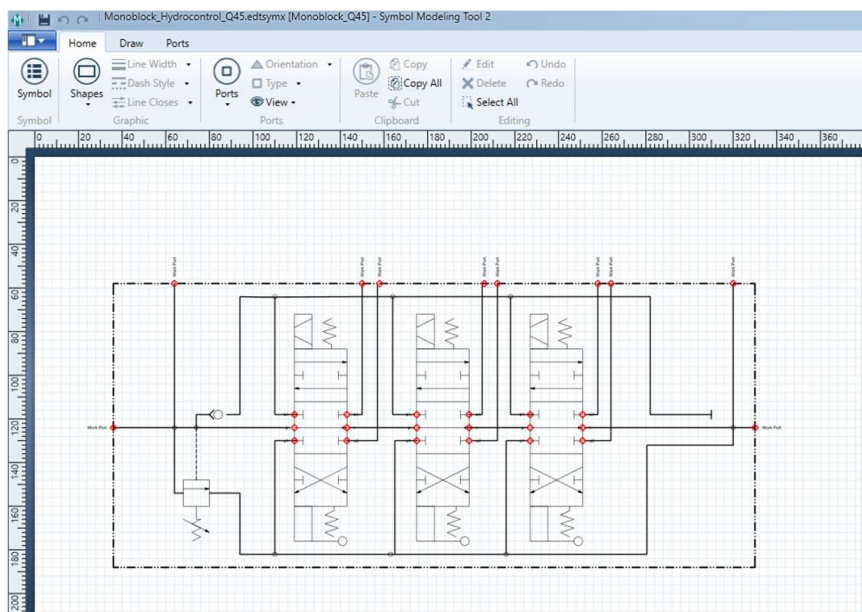
Kuva 43. Puomiston geometria, kun ollaan koukun ja moduulin välisellä tartuntakorkeudella.

Toinen haaste tarkan ja pehmeän liikkeen aikaan saamiseksi on kippisylinteriin asennetun kuormanlaskuventtiilin vaadittava avautumispaine, kun liike aloitetaan

äärimmäisestä taka-asennosta. Hankalan lähtökulman, kuormanlaskuventtiilin vastapaine, kuorman voiman välitys kippisylinterin varren puolelta ja lavasta syntyvä lisäkuorma vaatii huomattavan paineen, nostoliikkeen alkuvaiheessa, joka nopeasti pienenee geometrian muuttuessa edullisemmaksi.

Koska laitteen oma hydraulikkaventtiiliryhmä on päälle tai pois kytkeytyvän tyyppinen Monoblock, säätöä ei ole mahdollista suorittaa portaattomasti ilman lisäkomponenttien hankintaa ja asentamista. Ratkaisu ongelmaan löytyy alustan omasta PWM-ohjatusta proportionaaliventtiilistä, joka toimii sekä säätimenä että ulosottona laitteiston paineistukselle. Sekvenssien eri vaiheissa tarvittavat nopeudet säädetään ohjelmallisesti alustalta tulevan venttiililohkon virtauksen avulla, kun Monoblock karat aktivoidaan.

Koukkulaitteen hydraulikasta ja toimilaitteiden anturoinnista on laadittu hydraulikka kaavio, Bosch Rexroth ilmaisella Scheme Editor 6.3 versiolla. Laitteistosta ei ollut saatavilla virallista hydraulikkakaaviota. Kirjastot eivät sisältäneet kaikkia tarvittavia piirrosmerkkiä. Monoblock Q45 3-karainen ohjausventtiili (kuva 44) ja säädettävä kuormanlaskuventtiilin piirrosmerkit luotiin Scheme Editor 6.3 sisältyvällä Symbol Modelling Tool 2.1-ohjelmalla.

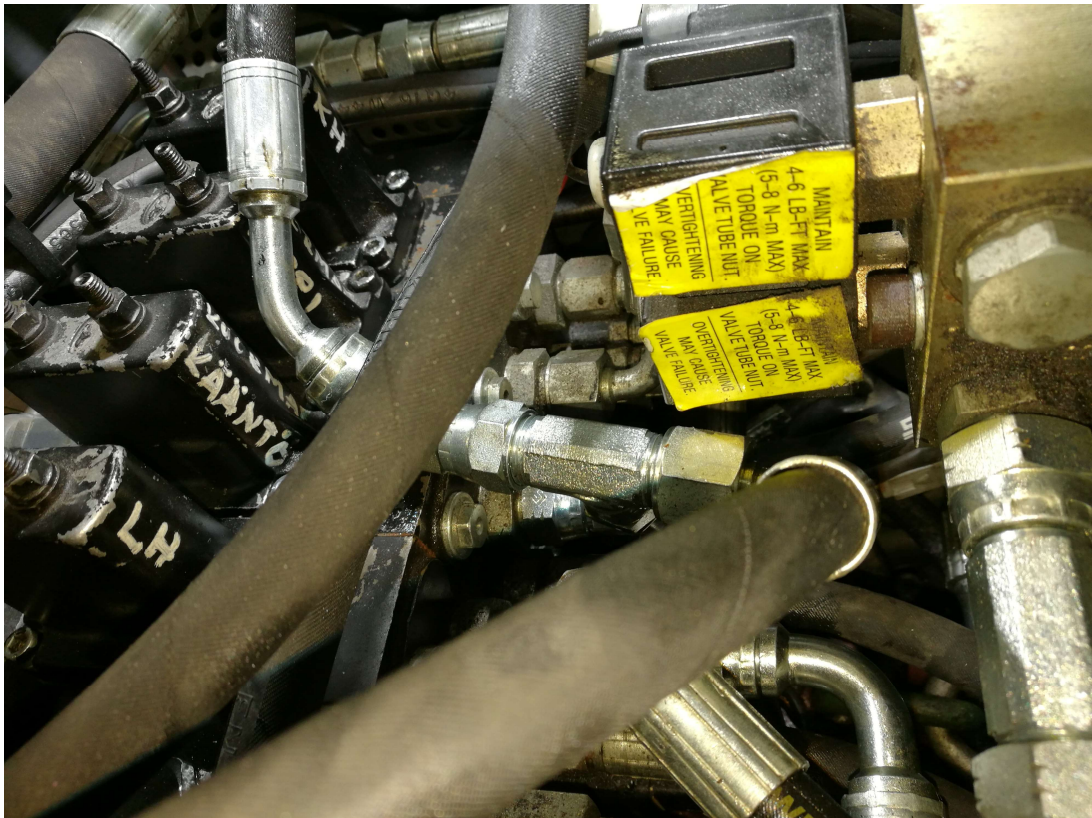


Kuva 44. Scheme Editor Modelling Tool 2.1-ohjelmalla luotu Monoblock Q45-venttiiliryhmän piirrosmerkki.

Luodut merkit tuodaan varsinaiselle kaavionpiirto ohjelmalle SVG formaatissa ja tallennetaan halutulle kirjastolle. Ohjelma on erittäin helppokäyttöinen ja sisältää tyypillisimmät, kuvan luontiin tarvittavat muodot ja komponenttien erikoismerkkejä. Hydraulikkakaavio on liitteenä 4.

9.2 Rakka 3000 UGV

Alustan hydraulikalta tarvittava painekytkentä on liitetty vapaana olevalle Walvoilin proportionaaliventtiilille ja tankkiliitäntä suoraan alustan hydraulikkasäiliölle menevälle kokoojalle, ennen paluusuodatinta.



Kuva 45. Walvoil proportionaaliventtiilit (PWM-ohjaus) rungon sisällä.

10 Sähkökytkennät

10.1 CPS-koukkulaite

Koukkulaitteelle tuotu 24 V:n syöttöjännite on kytketty 4-paikkaiselle sulakerasialle. Sulakerasialta jakaantuu syötöt erikseen IFM O3M-valaisinyksikölle, CR 1083-näyttömoduulille ja TEC132:lle. Näyttömoduulin ja O3M-anturin välille kytketty kaapeli toimii sekä CAN-signaalien välittämiseen, että jännitesyöttönä itse anturille. Technionin ja toimilaitteiden/antureiden väliset kytkennät ovat kytkentälistana (liite 5).

10.2 Rakka 3000 UGV alusta

Alustalta on kytketty päävirtakytkimen jälkeen 24 V:n syöttöjännite koukkulaitteelle asennetulle sulakerasialle. Alustan akulta on myös kytketty erillinen maajohdin koukkulaitteen runkoon, maadoituksen varmistamiseksi. CAN-kaapeli on kytketty alustan pääsähkökaappiin, liittimelle X7, josta pinnit 1&2 ovat syöttöjä, 3&4 maa, 5 CAN_H ja 6 CAN_L.

11 Lavan varustelu ja mekaaniset muutokset

Lavan etuseinään on asennettu kaksi kappaletta IFM:n valmistamia prismaattisia kolmioheijastimia. Heijastimet on asennettu maasta keskelle mitattuna 51 cm:n päähän, niiden etäisyys toisistaan on 71 cm keskeltä keskelle mitattuna (kuva 46).



Kuva 46. IFM E3M140-kolmioheijastinpari.

Koukkulaitteen koukun profiilia modifioitiin helpottamaan lavan tartuntaan kiinnittymistä ja irrottamista. Alkuperäinen profiili vaatii irrottamisvaiheessa sekä kipin että liukupalkin ajamista vuorotellen, kunnes tartunta ohittaa koukussa olevan lukituskyynen, jonka tarkoituksena on pitää lava lukittuna pituussuunnassa laitteen päällä. Materiaalia jyrrettiin pois kyynen päältä ja etupuolelta, millä saatiin tarvittava tila ja muoto helpottamaan koukun alta ajoa (kuva 47).



Kuva 47. Koukun profiilin muokkaus, vasemmalla näkyy alkuperäinen profiili ja oikealla modifioitu profiili.

Profiilia korjattiin myös koukun tartuntapuolelta, niin että lavaa lasiessa moduulin tartunta liukuu jo juuri ennen lavan maakosketusta taaksepäin 1 cm aiempaa enemmän ja jättää tartunnan ja koukun välisen aseman lukituskyynen takapuolelle, helpommin irrotettavaksi.

12 Tulosten arviointi ja jatkokehitysehdotukset

12.1 Koukkulaitteen automatisointi

Koukkulaitteen nosto- ja laskusekvenssien ja manuaaliohjauksen ohjelmointi toimii onnistuneesti, halutuilla ominaisuuksilla. Ohjelmoinnissa käytettiin sekä FBD- että ST-ohjelmointikieliä monipuolisesti ja tilanteisiin sopivilla tavoilla. Autonomisesti ohjaavien ohjelmat on rajoitettu toimimaan yhden työkierron läpi, oletuksella että moduuli on kytkeytynyt onnistuneesti nostosyklille ja päälle tulo on onnistunut. Vastaavasti laskuoperaatiossa lava on laskettu ja onnistuneesti irronnut lavan koukkutartunnasta. Manuaaliohjaukset toimivat samalla tarkkuudella kuin aiemmin, sillä erotuksella että luodulla funktioblokilla, joka säätää PWM-arvoa on saatu aikaiseksi rauhallisemmat liikkeet aloituksessa.

Anturoinnin valinta oli laitteen muille toiminnoille onnistunut paineanturia lukuun ottamatta, jota käytettiin moduulin ja koukun välisen kytkeytymisen todentamiseen. Kippisylinterin kuormanlaskuventtiilistä johtuen, tarkan minimipaineen raja-aluetta on vaikea määritellä kevyellä kuormalla verrattuna tyhjään puomiin. Punnitusnoston jälkeen on pidettävä ajastimella minimissään 10 s:n tauko, jolloin paine tasoittuu anturilla ja saa luotettavan lukeman. Paineen tasoituksen tarve johtuu paineanturin kytkentäpaikasta sylinterin varren kammion linjassa lukkoventtiilin jälkeen. Liikkeen käytön aikana vaaditaan painetason ylitys, joka avaa ensin lukkoventtiilin ja mahdollistaa öljyn virtauksen venttiilin läpi ohjattavalle sylinterin kammiolle. Liikkeen lakattua lukkoventtiilin ja sylinterin kammion välille jää hienoinen ylipaine, joka hitaasti laskee ja tasoittuu tauon aikana.

Hydrauliikkapiiri toimii käytetyillä komponenteilla riittävän tarkasti. Pieniä viiveitä huomaa manuaaliajolla liikkeitä aloittaessa ja lopettaessa. Viive johtuu monoblockin karojen keskiasentoa palauttavien jousien voimasta ja ohjauskelojen vetovoimasta sekä voimanotosta alustan PWM-venttiilin välityksellä.

12.2 Moduulin paikannus ja UGV:n asemointi

Lavan paikannus heijastimia ja AGV Follow-ohjelmaa hyödyntämällä voidaan todeta toimivaksi ratkaisuksi. Rinnakkain toimivan ifm_O3M_OD_V041906 Object_List-

ohjelmasta hyödynnetty visuaalinen käyttöliittymä toimii apuna anturin havaintoalueella olevista kohteista testausvaiheessa. Anturin asennuskorkeuteen ja asennuskulmaan vaikuttivat heijastimien asennuskorkeus ja sijainti lavassa. Anturin asennuskorkeus laskettiin alkuperäisestä 1 000 mm:stä 930 mm:iin ja kallistukseksi testien perusteella jäi 14 astetta. Testeissä käytettiin punaisia kolmio-, keltaisia pyöreä-, valkoisia suorakaideaajoneuvoheijastimia sekä IFM:n prismaattisia kolmioheijastimia. Lopullisessa kokoonpanossa IFM:n oma heijastin toimi parhaiten pidemmillä etäisyyksillä fyysisen kokonsa ja hyvien prismaattisen heijastusominaisuuksiensa ansiosta.

Alustan lähestyessä lavaa, heijastinhavainto loppui 1,36–2,42 m:n välillä eri anturin asennuskorkeuksista ja kulmista riippuen. Anturin sijainnilla on korkeuden suhteen liikkumavaraa 400 mm. Anturin asennus liian alas altistaisi sen likaantumiselle ja maastossa olevalle aluskasvillisuuden osumille, josta päädyttiin kompromissiin rajoittaa anturin havaintoaluetta ja kääntää kuvauskulma mahdollisimman lähelle ajoneuvoa kuvaamaan maata. Heijastimien havaitsemisen loppumiseen vaikutti maareferenssin loppuminen skannerin keilausalueelta.

Kallistamalla anturia alaviistoon 14 astetta, lyheni havaintopisteen matka lopuksi keskimäärin 1,4 m:iin ja maksimi havaitsemismatka 40 cm:n korkeudelta mitattuna 10 m:iin. Heijastinpari on asennettu lavan syvennysosaan, jossa se on suojassa törmäyksiltä ja likaantumiselta. Asennusväli heijastimen keskeltä keskelle on 710 mm ja asennuskorkeus keskelle heijastinta 530 mm. Anturin parametrisoinnissa on määritetty havaittavien heijastavien objektien minimi korkeudeksi 400 mm. IFM:n prismaattisen kolmioheijastimen fyysisillä mitoilla jää sen alareunan mitaksi maapinnasta 450 mm.

Alustan paikoitustarkkuus anturin mittaustiedon ja sen perusteella annetuilla hydraulikkaventtiilien ohjausarvoilla on tässä projektissa käytetyllä tekniikalla kiitettävän tarkkaa +/- 2 cm koukun ja tartunnan sivuttaissuunnassa keskilinjalla. Lopulliseen sivusuunnan tarkkuuteen vaikuttaa viimeinen 42 cm:n peruutusmatka, joka suoritetaan pelkällä etäisyystiedolla. Tartunnan ja koukun fyysiset ominaisuudet sallisivat jopa +/- 10 cm poikkeaman sivusuunnassa onnistuneelle tartunnalle.

Pituussuunnassa päästiin aivan yhtä hyviin tuloksiin +/- 2 cm. Virheeseen vaikuttaa ajoneuvon lähestymisnopeus, jarrujen kytkeytymisnopeus ja saadun mittaustiedon tarkkuus, kun pysäytyskomento on suoritettu. Pituussuuntainen virhe huomioidaan koukun kiinni ajamisen yhteydessä, mittaamalla lavan etuseinän ja alustan todellinen

asema. Saadusta mitasta vähennetään puomin pituus ja ajetaan liukupalkilla koukku tartuntaan. Tässäkin tapauksessa on moduulin paikasta riippuen mahdollista onnistua tarvittaessa tarttumaan kiinni jopa 40 cm:n päästä.

12.3 Jatkokehitysehdotukset

Vastaavanlaisessa projektissa, jossa käytetään runko-ohjattua ajoneuvoa, jossa on hydraulinen nelipyöräveto, olisi kokeilemisen arvoista suorittaa laitteen ohjaaminen kohteelle vapauttamalla runko-ohjauksen sylinterit ns. kelluntaan ja säätää etummaisten pyörien vetomoottoreiden nopeuksia keskenään. Näin jäisi pois monimutkainen käännönohjauksen ja vetomoottoreiden pyörimisnopeuksien välinen laskenta. Tämä vaatisi hydraulikkamoottoreilta pyörimisnopeuden anturointia ja hydraulikkaa ohjaavalta lohkolta mahdollisuutta vapauttaa takimmaisten pyörien veto vapaalle. Ohjelmoinnissa tarvitsisi ainoastaan määrittää haluttu keskiajonoisuus ja laskea pyörien välinen nopeussuhde halutun käännöksen aikaan saamiseksi.

Koukkulaitteen hydraulikan ohjauksessa tulisi käyttää proportionaalisia LS-venttiileitä, ainakin kipin ja liukupalkin tarkkaan ohjaamiseen. Näille voisi helposti ohjelmaan liittää PID-säätimen, joka mahdollistaisi kuormasta riippumattonta nopeutta jo olemassa olevien antureiden antamien tietojen välityksellä esimerkiksi kulmanopeutta valvoen ja säätäen. Codesys 2.3-versiossa ei ole ainakaan Technionin puolesta valmiita ns. Motion Control kirjastoja kehitelty, joita voisi hyödyntää kyseiseen tarkoitukseen, joten aikaperusteiseen säätämiseen pitäisi luoda oma FB tai etsiä sopiva kolmannen osapuolen kirjasto. Moduulin kiinnittymisen koukkuun varmistamiseksi voisi kokeilla venymäänturia liukupalkin sisäputkessa paineanturin sijaan.

13 Yhteenveto

Kokonaisuutena projektia voidaan pitää varsin onnistuneena. Laitteen paikoitustarkkuus käytettävällä alustalla, riippumatta nopeuden tai kuljetun matkan valvonnasta oli suorastaan erinomainen. Koukkulaitteen fyysinen automatisointi sujui varsin mutkattomasti käytettävillä komponenteilla. Ohjelmointi Codesys 2.3-versiolla oli antoisaa ja välillä haastavaa johtuen sen rajallisuuksista verrattuna uudempiin 3.5-versioihin, joista on aiemmin kertynyt kokemusta. Pidän kuitenkin vanhemman version tuomista uusista kokemuksista logiikkaohjelmoinnin perustaitoja kehittävänä ja myös eräänlaisena perusohjelmoinnin kulmakivenä, jossa ohjelmoijan vastuulle jää kehittää omia funktioita ja oppia hyödyntämään rajallisimpia kirjastoja töissään. Ohjelmoinnin tukena käytin teosta Dag H. Hanssen, Programmable Logic Controllers, A Practical Approach to IEC 61131-3 Using Codesys.

Ohjelmointityön aikana ilmeni ongelma, joka vaikutti satunnaisesti alustan käännön kytkeytymiseen hetkellisesti (1 ms). Ongelman ratkaisu löytyi alustan ja TEC132-logiikan välisestä CAN-väylän ylikuormittumisesta. Väylän kuormitus johtui TEC132:n 1 ms:n välein lähtevistä viestistä. Ongelman löysi ja ratkaisi Rakkatecin systeemiasiantuntija Jere Knuuttila CAN-väylän kuunteluun tarkoitetulla adapterilla ja ohjelmalla, jolla etsittiin kyseisen ohjauskomennon lähettäjä. Tästä sain arvokasta oppia ja tietoa selvittää eri kirjastojen tarjolla olevien vaihtoehtojen sopivuutta kyseisiin operaatioihin ja näille vaadittavia lisätoimenpiteitä. Tässä kyseisessä tapauksessa luotiin ehdot, joilla viesti lähetetään ainoastaan siinä tapauksessa, että jokin tieto on muuttunut ja väylän kuormitus putoaa normaalille tasolle.

Aikataulullisesti projektin varsinainen toteuttaminen eteni sille varatussa ikkunassa, vaikka aloitus olikin muutaman kuukauden myöhässä, alkuperäisestä sovitusta ajankohdasta. Projektin oli tarkoitus alkaa aiemmin syksyllä 2017, mutta se viivästyi koneen varaustilanteen vuoksi, jolloin se oli työtehtävissä. Alusta vapautui joulukuun lopulla koukkulaitteineen, jolloin oli mahdollista aloittaa varustelu ja testaus. Ennen fyysisen alustan saapumista projektissa käytetyt komponentit, anturikamera ja logiikkojen kokoonpano koottiin Rakkatecin omissa tiloissa, joissa testattiin IFM-apuohjelmien toimivuutta ja oman ohjelmointityön linkitystä näille.

Koukkulaitteen ohjelmointityö tehtiin osittain ennakoon ja se kehittyi testien edetessä fyysisen laitteen ja käsiteltävän lavan kanssa. Pieniä komplikaatioita syntyi, kun alustan

alkuperäinen ohjelma välitti edelleen ohjauskomentoja PWM-venttiileille rinnakkain CAN-väylän yli TEC132:lle annettujen ohjauskomentojen kanssa. Suoranaisia päällekkäisiä ohjauskomentoja venttiileillä ei ollut, mutta komentoja välittyi alustan ohjausventtiileille, jotka oli poistettu koukkulaitteen ohjaamiselta. Venttiilit olivat tulpattuna ja ohjauskomennot aiheuttivat LS-piirissä täyden paineen pyynnön toimilaitepumpulta, aiheuttaen kaikissa rinnakkaisesti samaan aikaan toiminnassa olevissa piireissä ohjauksen maksimi paineella. Ongelma löytyi lopuksi kytkemällä hydraulikkapainemittari pääpainepiiriin ja kytkemällä toimintoja yksitellen päälle niin, että koukkulaitteen omat ohjausventtiilien sähköpistokkeet olivat irrotettuna.

Lopullisessa kokoonpanossa on yhdistettynä Rakkatec 3000:n alusta, jonka alkuperäistä logiikan koodia on muokattu vastaanottamaan ja välittämään toimilaitteille autonomiseen ohjaukseen tarvittavat signaalit, manuaaliohjauksen lisäksi, ja voidaan edelleen operoida alkuperäisellä kokoonpanolla ilman koodin muutoksia. IFM O3M-älykamera ja CR 1083-näyttömoduuli toimivat itsenäisenä kokonaisuutena, joka voidaan siirtää sellaisenaan toiselle alustalle tarvittaessa ja välittää CAN-viestein prosessoitua dataa. CPS:n antureilla varusteltu koukkulaite ja Technion TEC132-logiikkaohjain niin ikään ovat oma kokonaisuutensa, joka voidaan myös siirtää omana kokoonpanonaan mille alustalle tahansa, josta on saatavana 24 Voltin syöttöjännite ja 260 bar hydraulikkapaine. Projektista syntyi monipuolinen kokonaisuus, jossa sovellettiin koneautomaation eri osa-alueita innovatiivisesti, ja lopputulos vastasi tilaajan tarpeita.

Lähteet

- 1 Dechow, David. 2017. T1 The Fundamentals of Machine Vision. PDF-dokumentti. <<https://www.visiononline.org/userAssets/aiaUploads/file/T1-The-Fundamentals-of-Machine-Vision.pdf>>.
- 2 Cognex. 2017. Machine Vision 2D & 3D Vision Systems, Vision Sensors. PDF-dokumentti. <https://www.cognex.com/downloads/literature-main?event=649fa31e-1838-4714-8932-c455a3cd5a69&cm_campid=273ca1b9-68d4-e311-a12c-5ef3fcdaf1b7>.
- 3 Cognex. 2016. Introduction to Machine Vision, A guide to automating process & quality improvements. PDF-dokumentti. https://www.assemblymag.com/ext/resources/White_Papers/Sep16/Introduction-to-Machine-Vision.pdf.
- 4 Martin, Daryl. 2012. A Practical Guide to Machine Vision Lighting. PDF-dokumentti. <<https://www.1stvision.com/elearning/Machine-Vision-Lighting-Tutorial.pdf>>.
- 5 Martin, Daryl. 2017. Beginning Lighting for Machine Vision. PDF-dokumentti. <https://www.visiononline.org/userAssets/aiaUploads/file/CVP_Beginning-Lighting-for-Machine-Vision_Daryl-Martin.pdf>.
- 6 Opto Engineering. 2017. Basics Opto Engineering 2017 EN. PDF-dokumentti. <https://www.opto-engineering.com/media/downloads/docs/Basics_Opto-Engineering_2017_EN.pdf>.
- 7 Filtering in Machine Vision. Verkkoaineisto. Edmund Optics. <<https://www.edmundoptics.com/resources/application-notes/imaging/filtering-in-machine-vision/>>. Luettu 30.3.2018.
- 8 Davies, E.R. 2012. Computer & Machine Vision, Theory, Algorithms, Practicalities. 4th ed. London. University of London.
- 9 Thete, Vinod. 2014. Blob-detection-pdf-dokumentti. Scribd. <<https://www.scribd.com/document/243398475/Blob-detection-pdf>>.
- 10 Luo, Daisheng. 1998. Pattern Recognition and Image Processing. E-kirja. Scribd. <<https://www.scribd.com/book/282656461/Pattern-Recognition-and-Image-Processing>>.
- 11 Understanding 3D Scanning. Verkkoaineisto. Autodesk. <<http://www.instructables.com/lesson/Understanding-3D-Scanning/>>. Luettu 30.3.2018
- 12 Texas Instruments. 2014. Verkkoaineisto. Time of Flight Camera – An Introduction. <<http://www.ti.com/lit/wp/sloa190b/sloa190b.pdf>>. Luettu 30.3.2018

- 13 Ringbeck, Hagebeuker. 2007. A 3D Time Of Flight Camera For Object Detection. PDF-dokumentti. <https://www.ifm.com/obj/O1D_Paper-PMD.pdf>.
- 14 Heredia, Conde, M. 2017. Phase-Shift-Based Time-of-Flight Imaging Systems. PDF-dokumentti. <http://www.springer.com/cda/content/document/cda_downloaddocument/9783658180560-c2.pdf?SGWID=0-0-45-1605415-p180831688>.
- 15 O3M_and_reflectors. 2017. Yrityksen sisäinen dokumentti. IFM. (Yrityksen sisäinen materiaali, ulkopuolisilla ei pääsyä)
- 16 IFM. 2018. Verkkoaineisto. O3M 3D Smart Sensor, Overview. <<https://www.ifm.com/ifmmy/web/mobile-3d.htm>>. Luettu 22.4.2018
- 17 ifm_O3M_FeaturePack_GlobalInformation. 2017. Yrityksen sisäinen dokumentti. IFM. (Yrityksen sisäinen materiaali, ulkopuolisilla ei pääsyä)
- 18 IFM. 2017. ifm O3M VisionAssistant Manual 706424 EN.PDF. PDF-dokumentti. <<https://www.ifm.com/mounting/706424UK.pdf>>.
- 19 IFM. 2016. IFM O3M251 ShortInstruction. PDF-dokumentti. <<https://www.ifm.com/mounting/80258595UK.pdf>>.
- 20 IFM. 2016. Devive manual, Inclination sensor 2 axes. PDF-dokumentti. <<https://www.ifm.com/mounting/80237498UK.pdf>>.
- 21 Technion. 2015. tec132-technical-manual-v1.1. PDF-dokumentti. <<http://technion.fi/wp-content/uploads/tec132-technical-manual-v1.1.pdf>>.
- 22 IFM. 2015. CR1083 installation instructions.PDF. PDF-dokumentti. <<https://www.ifm.com/mounting/7390934UK.pdf>>.
- 23 Description CODESYS AGV Following Sample Project.PDF. 2014. Yrityksen sisäinen dokumentti. IFM. (Yrityksen sisäinen materiaali, ulkopuolisilla ei pääsyä)

Ohjelmointityössä käytettyjä linkkejä

IFM Visual Assistant ohjelma versio 1.7.10 (julkaistu 8.9.2017) ja Firmware OD_4.19.6

<https://www.ifm.com/ifmus/web/o3m-download.htm>

CANfox Driver ja kallistusanturin konfigurointiin tarvittavat lataukset

<https://ifm.com/de/en/downloadarea/R360Content>

IFM I/O Link Master laitteen ajuri ja Firmware ovat ladattavissa tuotteen sivulta

<https://www.ifm.com/gb/en/product/E30390>

IFM I/O Link anturikohtaiset lisäohjelmat, ovat ladattavissa

<https://www.ifm.com/gb/en/downloadarea/IOContent>

IFM CR 1083 Module retrieve drive program

```

0001 (* Distance to retrievable module is controlled via camera measurement value*)
0002
0003 i_Distance_cm := REAL_TO_INT (prgAgvFollowing.AGV_Distance*100.0); (* value is scaled to cm*)
0004 i_Distance_MOD := (i_Distance_cm MOD 100);
0005 i_Distance_m := ((i_Distance_cm-i_Distance_MOD)/100);
0006 (* i_Distance_cm is calculated by dividing back to metres, Modulus is left over in cm*)
0007 (* i_Distance_cm is subtracted by the previous Modulus, leaving a even number in*)
0008 (*100's and then divided by 100 scaling back to metres*)
0009
0010 DISTANCE_CM := INT_TO_USINT (i_Distance_MOD);
0011 DISTANCE_M := INT_TO_USINT (i_Distance_m);
0012
0013 i_Displace := REAL_TO_INT (prgAgvFollowing.AGV_Displacement*100);
0014
0015 IF prgAgvFollowing.AGV_Displacement < 0 AND prgAgvFollowing.Obj_in_Yellow_Area = TRUE AND prgAgvFollowing.AGV_Detected = FALSE THEN
0016   x_NEG_DISPLACEMENT := TRUE;
0017   i_Displace_cm := i_Displace MOD 100*-1;
0018   i_Displace_m := (i_Displace - i_Displace_cm)/100*-1;
0019
0020 ELSEIF prgAgvFollowing.AGV_Displacement >= 0 THEN
0021   x_NEG_DISPLACEMENT := FALSE;
0022   i_Displace_cm := i_Displace MOD 100;
0023   i_Displace_m := (i_Displace - i_Displace_cm)/100;
0024
0025 ELSEIF (prgAgvFollowing.AGV_Displacement = -1 AND prgAgvFollowing.Obj_in_Yellow_Area = FALSE) OR
0026   (prgAgvFollowing.AGV_Displacement = -1 AND prgAgvFollowing.Obj_in_Yellow_Area = TRUE AND prgAgvFollowing.AGV_Detected = TRUE ) THEN
0027   x_NEG_DISPLACEMENT := FALSE;
0028   i_Displace_cm := 0;
0029   i_Displace_m := 0;
0030   i_Distance_cm := 0;
0031   i_Distance_m := 0;
0032   DISTANCE_M := 0;
0033 END_IF
0034
0035 DISPLACE_CM := INT_TO_USINT (i_Displace_cm);
0036 DISPLACE_M := INT_TO_USINT (i_Displace_m);

```

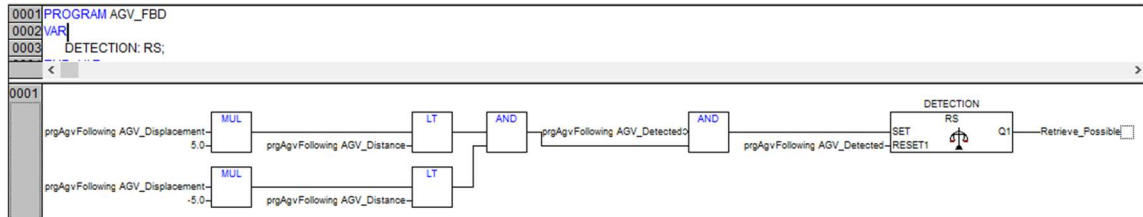
Havaitun kohteen etäisyyss- ja poikkeamatietojen skaalaukset ja konversiot. Aseman tietojen nollaus, kun Agv Following-ohjelma on menettänyt lukituksensa heijastinpariin.

```

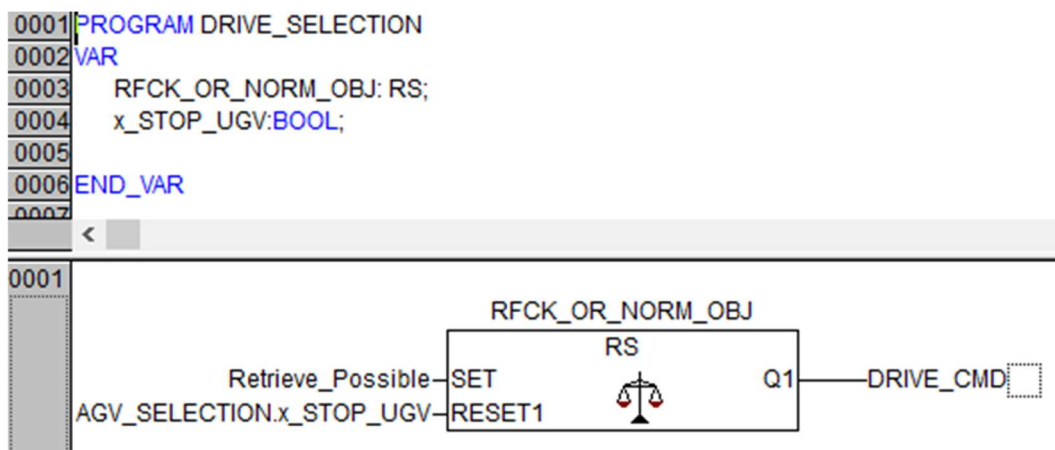
0001 PROGRAM AGV_2
0002 VAR
0003   x_OBJECT_DISTANCE :BOOL;
0004 END_VAR
0005
0006 <
0007
0008 (* After prgAgvFollowing has lost reflector recognition at close range, AGV_SELECTION program switches to measuring*)
0009 (* NORMAL_OBJECT_Distance between the given parameters *)
0010
0011 NORMAL_OBJECT_Distance := INT_TO_USINT ( ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_x1_cm);
0012
0013 IF ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_x1_cm > 106 AND
0014 ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_x1_cm <= 150 THEN
0015   x_OBJECT_DISTANCE := TRUE;
0016   DISPLACE_CM := 0;
0017   DISPLACE_M := 0;
0018   DISTANCE_CM := 0;
0019   DISTANCE_M := 0;
0020   DRIVE_CMD := 1;
0021
0022 ELSEIF ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_x1_cm >= 100 AND
0023 ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_x1_cm <= 106 THEN
0024   x_OBJECT_DISTANCE := FALSE;
0025   DRIVE_CMD := 0;
0026 END_IF

```

Ohjauskomento ja asematietojen pakottaminen nolaksi lavan lähestymisen kakkosvaiheessa ilman heijastintunnistusta.



Ehdot havaitun kohteen noutamiseksi. Etäisyyden ja poikkeaman suhteen on oltava alle 5:1.



Ajokomennon kytkeminen ja resetointi.

0001	PROGRAM AGV_SELECTION
0002	<
0003	x_MODULE_TRUE := FALSE;
0004	x_OBJECT_DISTANCE := FALSE;
0005	x_STOP_UGV := FALSE;
0006	NORMAL_OBJECT_DISTANCE := INT_TO_USINT(ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_x1_cm);
0007	i_WIDTH_OBJ_72 := ((ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_y1_cm * -1) +
0008	ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_y2_cm);
0009	DRIVE_SELECTION();
0010	
0011	IF ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_x1_cm >= 100 AND
0012	ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_x1_cm <= 150 AND
0013	ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].by_id = 72 AND
0014	ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_x2_cm >=
0015	(ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_x1_cm + 1) AND
0016	prgAgvFollowing.AGV_Detected = TRUE AND
0017	i_WIDTH_OBJ_72 >= 20 THEN
0018	
0019	x_MODULE_TRUE := TRUE;
0020	AGV_2();
0021	x_AGV_1_PRG := FALSE;
0022	
0023	ELSIF ReceiveObjectInformationfromCAN.o_fb_CANReadObjectInfo.o_ast_o3mObject[0].i_x1_cm > 150 THEN
0024	
0025	
0026	
0027	PLC_PRG_AGV_Follow();
0028	prgAgvFollowing();
0029	AGV_1();
0030	AGV_FBD();
0031	x_AGV_1_PRG := TRUE;
0032	
0033	END_IF

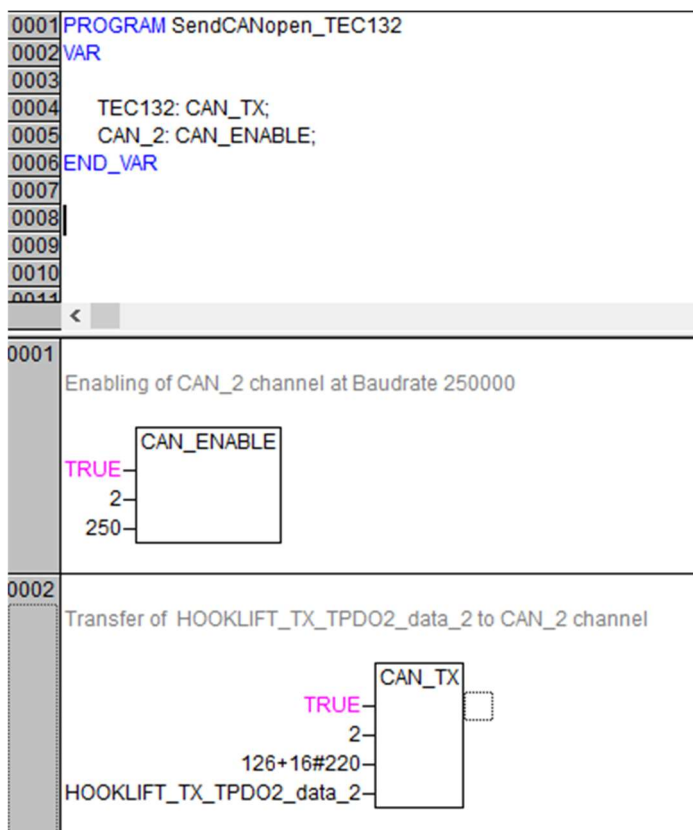
Noutamisen ohjelman valinta, perustuu havaitun heijastinparin asemaan tai havaitun normaalin objektin tunnisteeseen (ID) ja sen havaittujen mittojen vastaamiseen annettuja parametreja.


```

0001 VAR_GLOBAL
0002   HOOKLIFT_TX_TPDO2_data_2 AT %MB1344 : ARRAY[0..7] OF BYTE;
0003
0004   HOOKLIFT_TX_TPDO2_BYTE_0 AT %MB1344 : BYTE; (* command bits *)
0005   HOOKLIFT_TX_TPDO2_BYTE_1 AT %MB1345 : BYTE;
0006   NORMAL_OBJECT_Distance AT %MB1346 : USINT; (* When reflector Object is too near, switches to Normal Object in position Obj [0] cm_x_1*)
0007   DISPLACE_CM AT %MB1347 : USINT;
0008   DISPLACE_M AT %MB1348 : USINT;
0009   HOOKLIFT_TX_TPDO2_BYTE_5 AT %MB1349 : BYTE;
0010   DISTANCE_CM AT %MB1350 : USINT;
0011   DISTANCE_M AT %MB1351 : USINT;
0012
0013   HOOKLIFT_TX_TPDO2_WORD_0 AT %MW672 : WORD; (* command bits *)
0014   HOOKLIFT_TX_TPDO2_WORD_1 AT %MW673 : INT;
0015   HOOKLIFT_TX_TPDO2_WORD_2 AT %MW674 : INT;
0016   HOOKLIFT_TX_TPDO2_WORD_3 AT %MW675 : INT;
0017
0018
0019
0020   Retrieve_Possible AT %MX672.0: BOOL; (* Module in possible lift section*)
0021   x_NEG_DISPLACEMENT AT %MX672.1: BOOL; (* Displacement is to the right of center position, NEGATIVE DISPLACEMENT TRUE*)
0022   DRIVE_CMD AT %MX672.2: BOOL; (*Reverse drive =1*)
0023   HOOKLIFT_TX_bit3 AT %MX672.3: BOOL;
0024   HOOKLIFT_TX_bit4 AT %MX672.4: BOOL;
0025   HOOKLIFT_TX_bit5 AT %MX672.5: BOOL;
0026   HOOKLIFT_TX_bit6 AT %MX672.6: BOOL;
0027   HOOKLIFT_TX_bit7 AT %MX672.7: BOOL;
0028 END_VAR

```

Lähetettävien CAN viestien muuttujat.



Lähtettämiseen luodut CAN-lohkot.

Technion TEC 132 Hooklift and UGV steering program

```

0001 x_CAN_ENABLE := FALSE;
0002
0003 CAN_RX();
0004 Sensor_DATA();
0005 LINTRAFO_SENSORS();
0006 HOOKLIFT_CONTROL();
0007 DRIVE_TURN();
0008 CONTROL_CONDITIONS();
0009 PLC_Valve_CONTROL();
0010
0011 IF DATA_CHANGED() THEN
0012     x_CAN_ENABLE := TRUE;
0013     CAN_TX_1();
0014 END_IF
0015

```

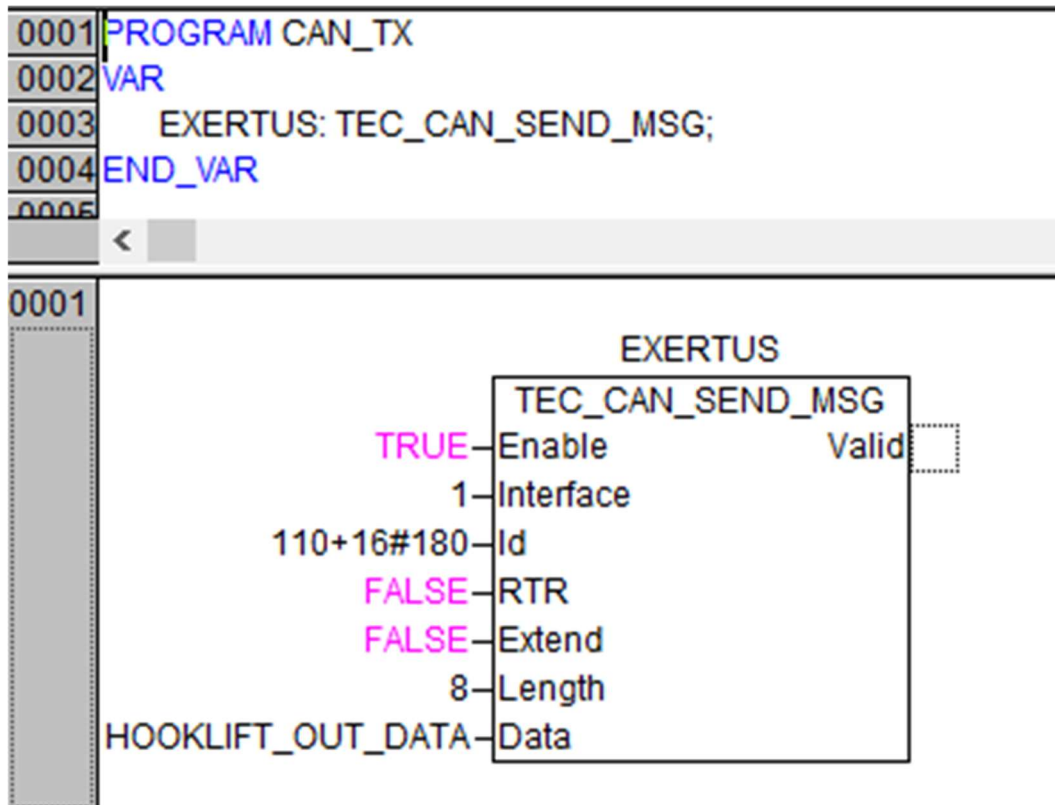
Pääohjelma PLC_PRG kutsuu ja suorittaa ohjelmat.

```

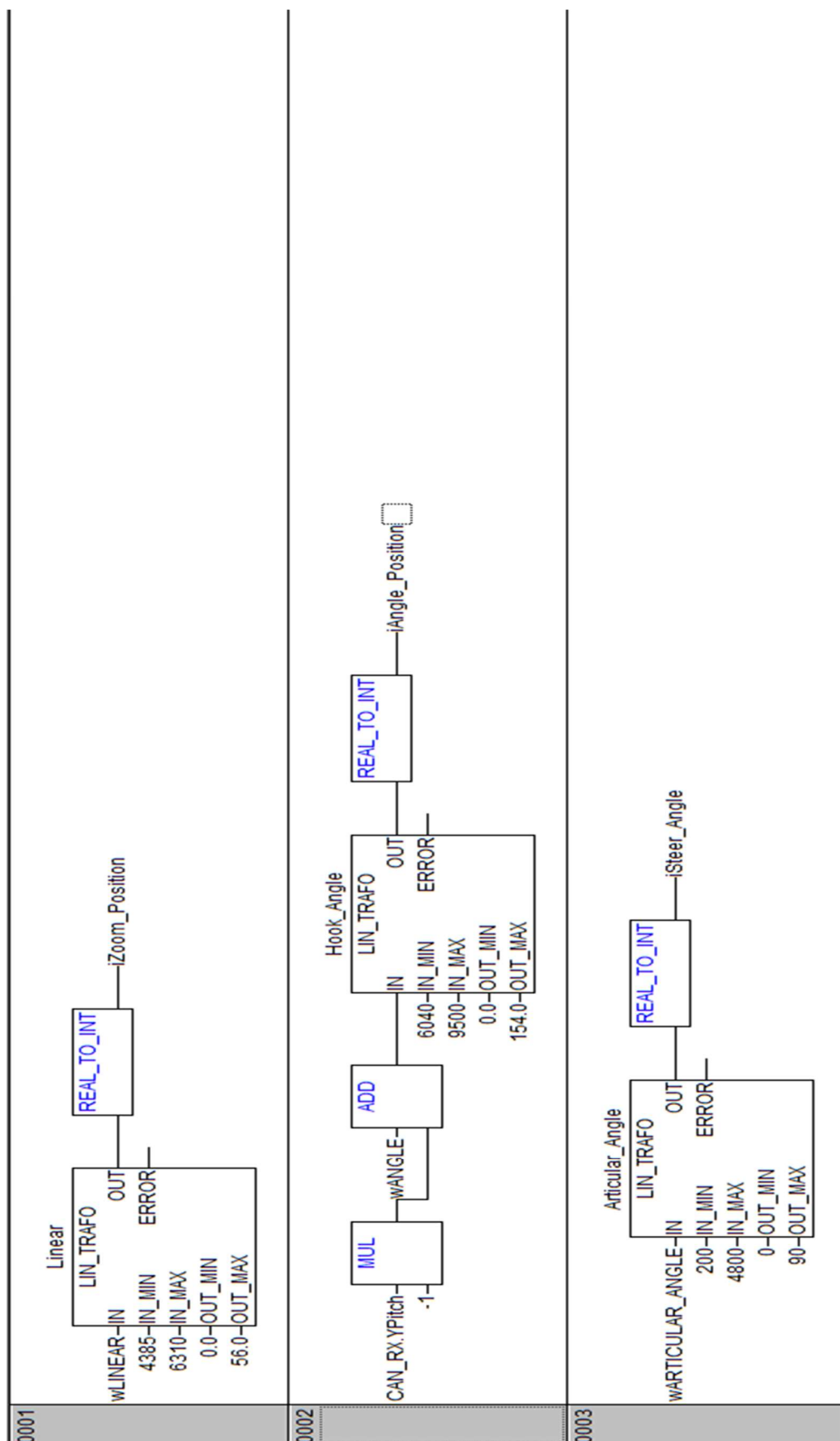
0001 Vehicle_CANbus();
0002 IFM_CANbus();
0003
0004 Sender_ID1 := Vehicle_CANbus.Id;
0005
0006 IF Sender_ID1 = 113+16#180 THEN
0007     Gyro_DATA := Vehicle_CANbus.Data;
0008     pYRoll := ADR(Gyro_DATA[0]);
0009     YRoll := (pYRoll^+YRoll_Correction)*0.0573; (*muutettuna asteiksi *0,573*)
0010     pYPitch := ADR(Gyro_DATA[2]);
0011     YPitch := (pYPitch^+YPitch_Correction)*0.0573;
0012
0013 ELSIF Sender_ID1 = 110+16#200 THEN
0014     HOOKLIFT_IN_DATA := Vehicle_CANbus.Data;
0015     ZoomCmd := ADR(HOOKLIFT_IN_DATA[3]);
0016     BoomCmd := ADR(HOOKLIFT_IN_DATA[2]);
0017     HookliftPWR := BITADR(%MX4.0);
0018     HookliftUNLD := BITADR(%MX4.1);
0019     HookliftLD := BITADR(%MX4.2);
0020     HookliftLOCK_OPEN := BITADR(%MX4.3);
0021     HookliftLOCK_CLOSED := BITADR(%MX4.4);
0022 END_IF
0023
0024 Sender_ID2 := IFM_CANbus.Id;
0025
0026 IF Sender_ID2 = 126+16#220 THEN
0027     IFM_DATA_1 := IFM_CANbus.Data;
0028 END_IF
0029

```

ST:llä luotu CAN_Rx.



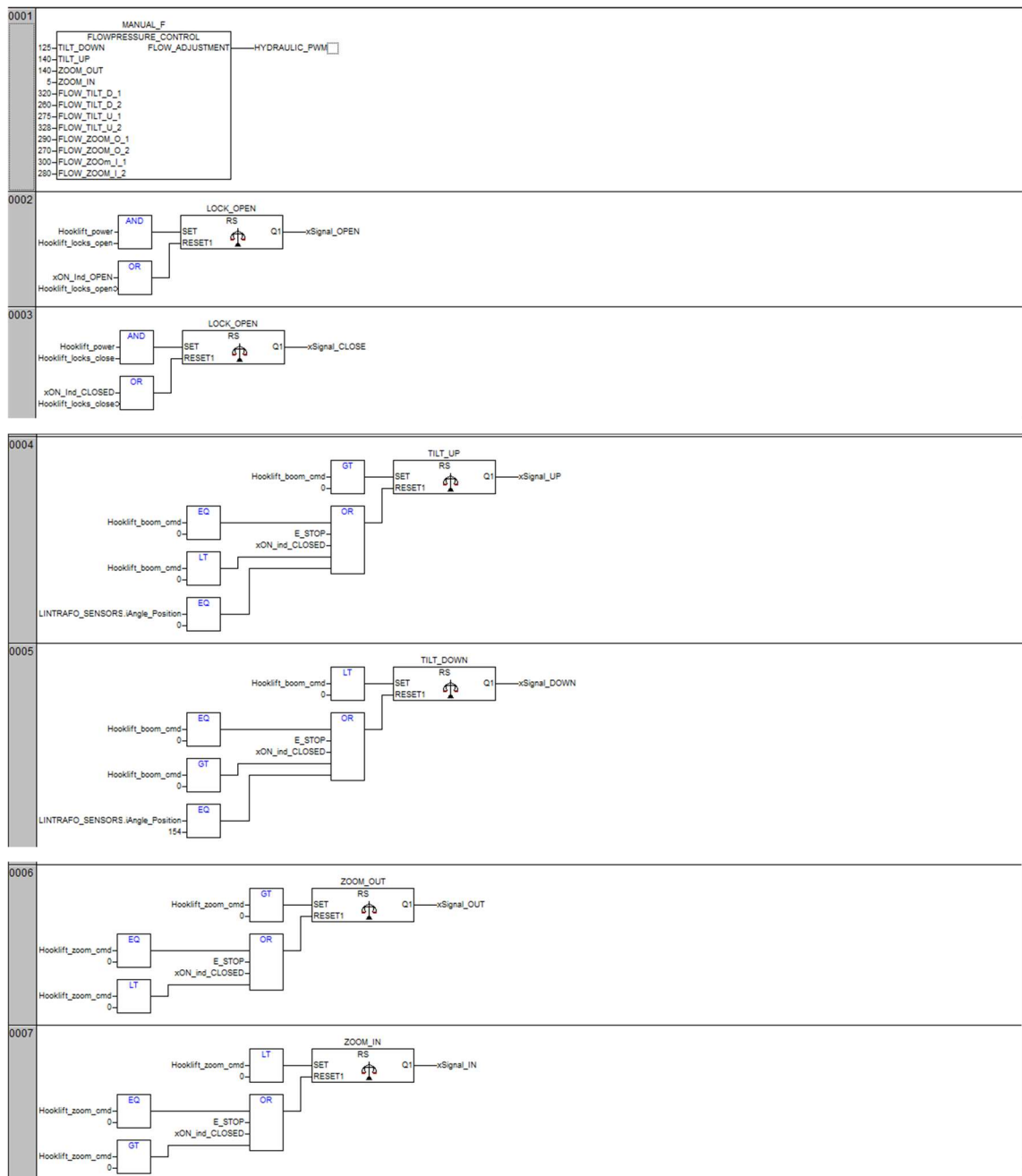
FBD:llä luotu CAN_Tx.



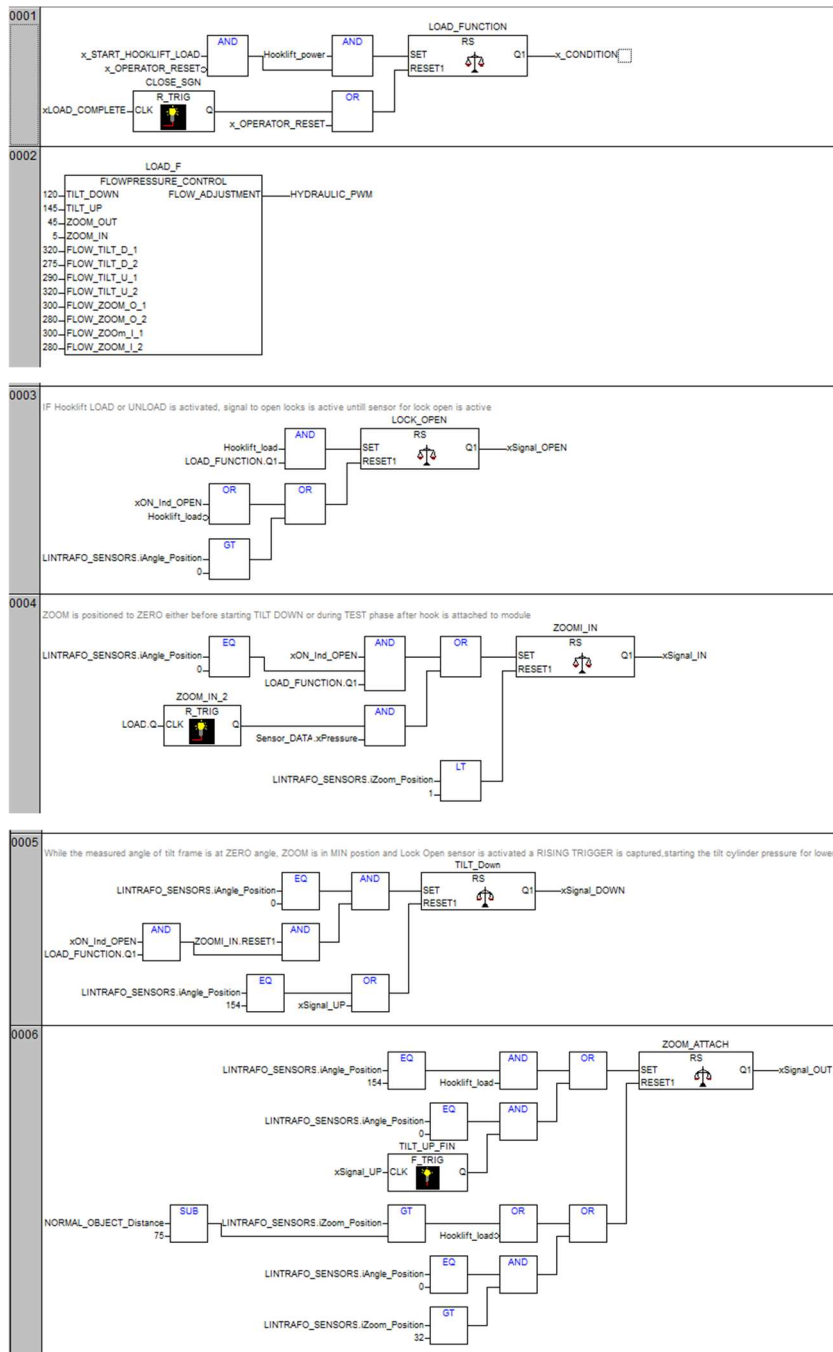
FBD:llä luotu, LIN_Trafo-lohkoilla antureiden skaalaukset.

```
0001 PROGRAM HOOKLIFT_CONTROL
0002
0003 IF Hooklift_load = TRUE AND Hooklift_unload = FALSE THEN
0004     HOOK_LOAD();
0005     x_Hook_load := TRUE;
0006     x_Hook_unload := FALSE;
0007     x_Manual := FALSE;
0008     Hooklift_operation_in_progress := TRUE;
0009     x_CAN_ENABLE := TRUE;
0010 ELSIF Hooklift_unload = TRUE AND Hooklift_load = FALSE THEN
0011     HOOK_UNLOAD();
0012     x_Manual := FALSE;
0013     x_Hook_load := FALSE;
0014     x_Hook_unload := TRUE;
0015     Hooklift_operation_in_progress := TRUE;
0016     x_CAN_ENABLE := FALSE;
0017 ELSIF Hooklift_load = FALSE AND Hooklift_unload = FALSE THEN
0018     HOOK_MANUAL();
0019     x_Manual := TRUE;
0020     x_Hook_load := FALSE;
0021     x_Hook_unload := FALSE;
0022     Hooklift_operation_in_progress := FALSE;
0023     x_CAN_ENABLE := FALSE;
0024 END_IF
```

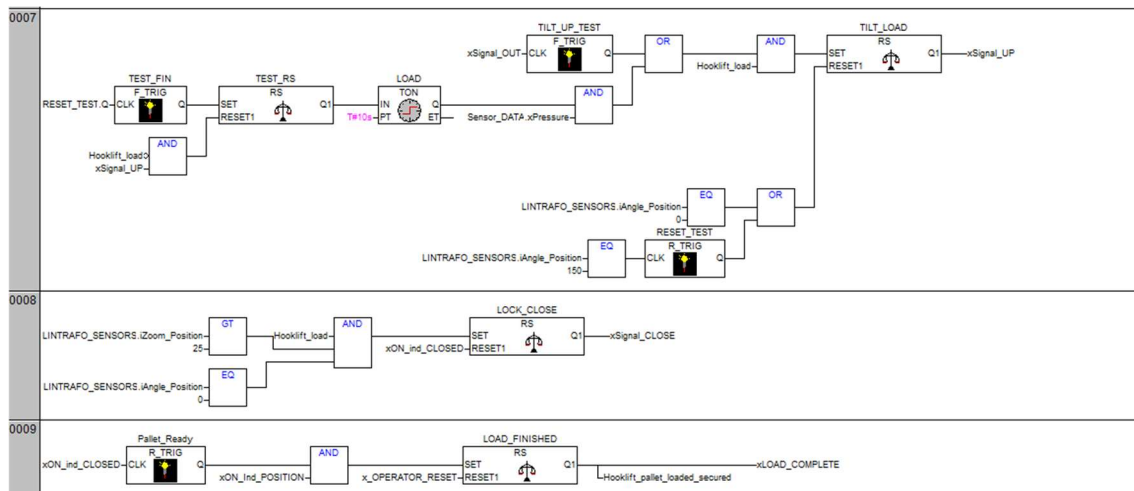
ST:llä luotu koukkulaitteen eri ohjausohjelmien toimintaehdot.



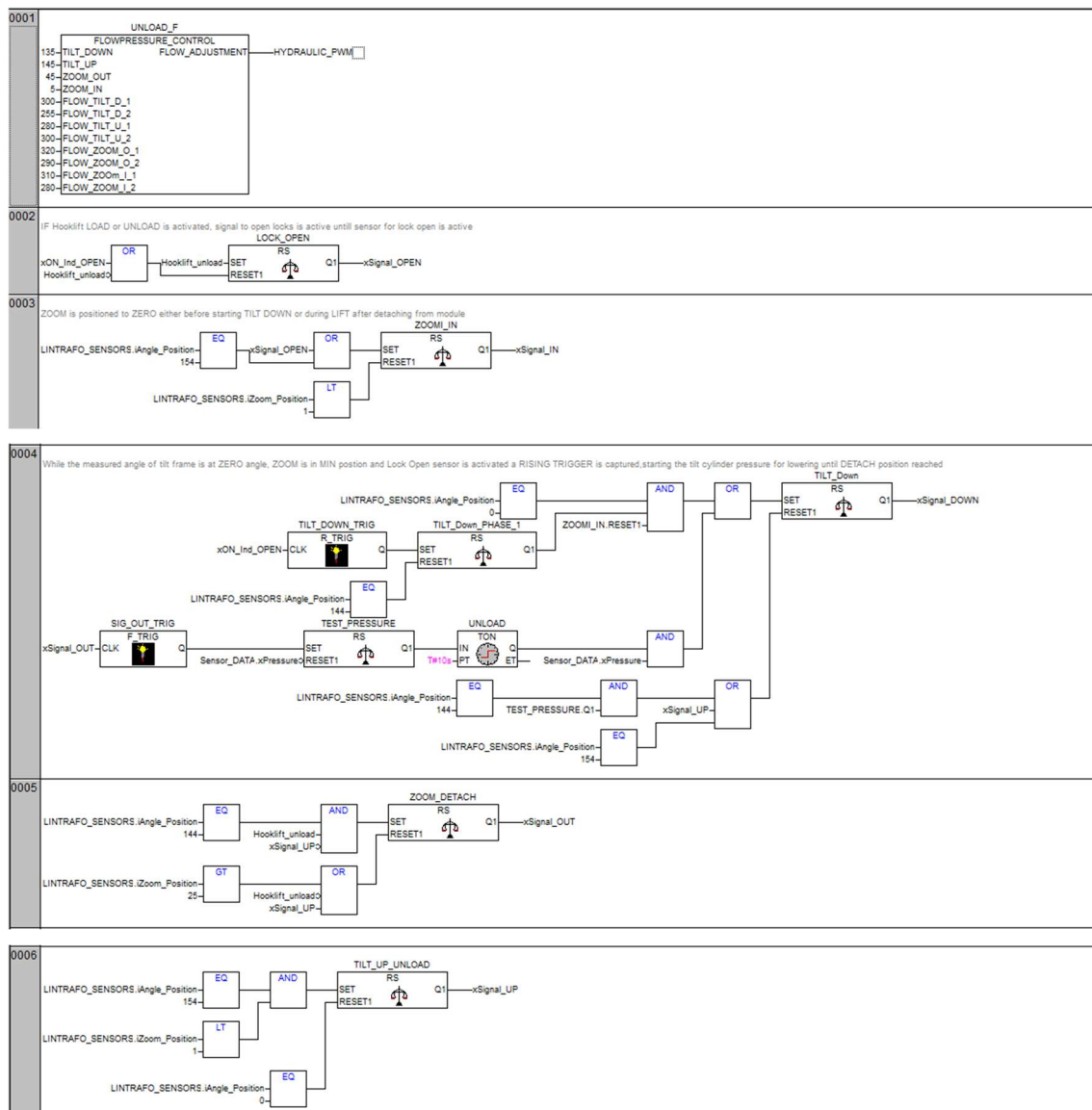
FBD:llä luotu koukkulaitteen manuaalisen ajon ohjaus (Manual).



FBD:llä luotu moduulin automaattinen lastaus-sekvenssi (Load) osa 1.



FBD:llä luotu moduulin automaattinen lastaus-sekvenssi (Load) osa 2.



FBD:llä luotu koukkulaitteen moduulin automaattinen laskusekvenssi (Unload).


```

0032
0033 rMeasuredPressurePlus:=WORD_TO_REAL(wPRESSURE_PLUS);
0034
0035 IF rMeasuredPressurePlus >= 1800 AND rMeasuredPressurePlus < 3500 THEN
0036     xPressure := TRUE;
0037     ELSE xPressure := FALSE;
0038 END_IF
0039
0040 rMeasuredPressureMinus:=WORD_TO_REAL(wPRESSURE_MINUS);
0041
0042 IF rMeasuredPressureMinus >= 4000.0 THEN
0043     xPressure_M := TRUE;
0044 END_IF
0045
0046 rArticularAngle := WORD_TO_REAL(wARTICULAR_ANGLE);
0047 Steering := REAL_TO_INT(rArticularAngle);
0048

```

```

0001
0002 ("Regulated 24 V voltage output/supply to sensors")
0003
0004 INDUCTIVE_SWITCH_POWER := 1000; ("inductive switches_earth connected to X2-29")
0005 PRESSURE_SWITCH_POWER := 1000; ("pressure switches_earth connected to X2-30")
0006 LINEAR_ANGLE_SENSOR_POWER := 1000; ("linear and angle sensors_earth connected to X2-28")
0007 ARTICULAR_SENSOR_POWER := 1000; ("hall-sensor power")
0008
0009 rMeasuredLinear := WORD_TO_REAL(wLINEAR);
0010 iLinear := REAL_TO_INT(rMeasuredLinear);
0011
0012 rMeasuredAngle:=WORD_TO_REAL(wANGLE);
0013 iAngle := REAL_TO_INT(rMeasuredAngle);
0014
0015 rON_Ind_OPEN:=WORD_TO_REAL(wInd_OPEN);
0016 IF rON_Ind_OPEN <= 5000 AND rON_Ind_OPEN > 500 THEN
0017     xON_Ind_OPEN := TRUE;
0018     ELSE xON_Ind_OPEN := FALSE;
0019 END_IF
0020
0021 rON_Ind_CLOSED:=WORD_TO_REAL(wInd_CLOSED);
0022 IF rON_Ind_CLOSED <= 8000 AND rON_Ind_CLOSED > 500 THEN
0023     xON_Ind_CLOSED := TRUE;
0024     ELSE xON_Ind_CLOSED := FALSE;
0025 END_IF
0026
0027 rON_Ind_POSITION:=WORD_TO_REAL(wInd_POSITION);
0028 IF rON_Ind_POSITION <= 8000 AND rON_Ind_POSITION > 1000 THEN
0029     xON_Ind_POSITION := TRUE;
0030     ELSE xON_Ind_POSITION := FALSE;
0031 END_IF

```

ST:llä luotu antureiden mittaus- ja toimintaehdot.

```

0001 HY_1 := wLock_OPEN;
0002 HY_2 := wLock_CLOSE;
0003 HY_3 := wTilt_UP;
0004 HY_4 := wTilt_DOWN;
0005 HY_5 := wTelescope_IN;
0006 HY_6 := wTelescope_OUT;
0007 HYDRAULIC_PWM := REAL_TO_WORD(rHydFlowAdjustment);
0008
0009 IF xSTOP = FALSE AND xSignal_UP = TRUE THEN
0010     wTilt_UP := 1000;
0011     ELSE wTilt_UP := 0;
0012 END_IF
0013
0014 IF xSTOP = FALSE AND xSignal_DOWN = TRUE THEN
0015     wTilt_DOWN := 1000;
0016     ELSE wTilt_DOWN := 0;
0017 END_IF
0018
0019 IF xSTOP = FALSE AND xSignal_IN = TRUE THEN
0020     wTelescope_IN := 1000;
0021     ELSE wTelescope_IN := 0;
0022 END_IF
0023
0024 IF xSTOP = FALSE AND xSignal_OUT = TRUE THEN
0025     wTelescope_OUT := 1000;
0026     ELSE wTelescope_OUT := 0;
0027 END_IF
0028
0029 IF xSTOP = FALSE AND xSignal_OPEN = TRUE THEN
0030     wLock_OPEN := 1000;
0031     ELSE wLock_OPEN := 0;
0032 END_IF
0033
0034 IF xSTOP = FALSE AND xSignal_CLOSE = TRUE THEN
0035     wLock_CLOSE := 1000;
0036     ELSE wLock_CLOSE := 0;
0037 END_IF

```

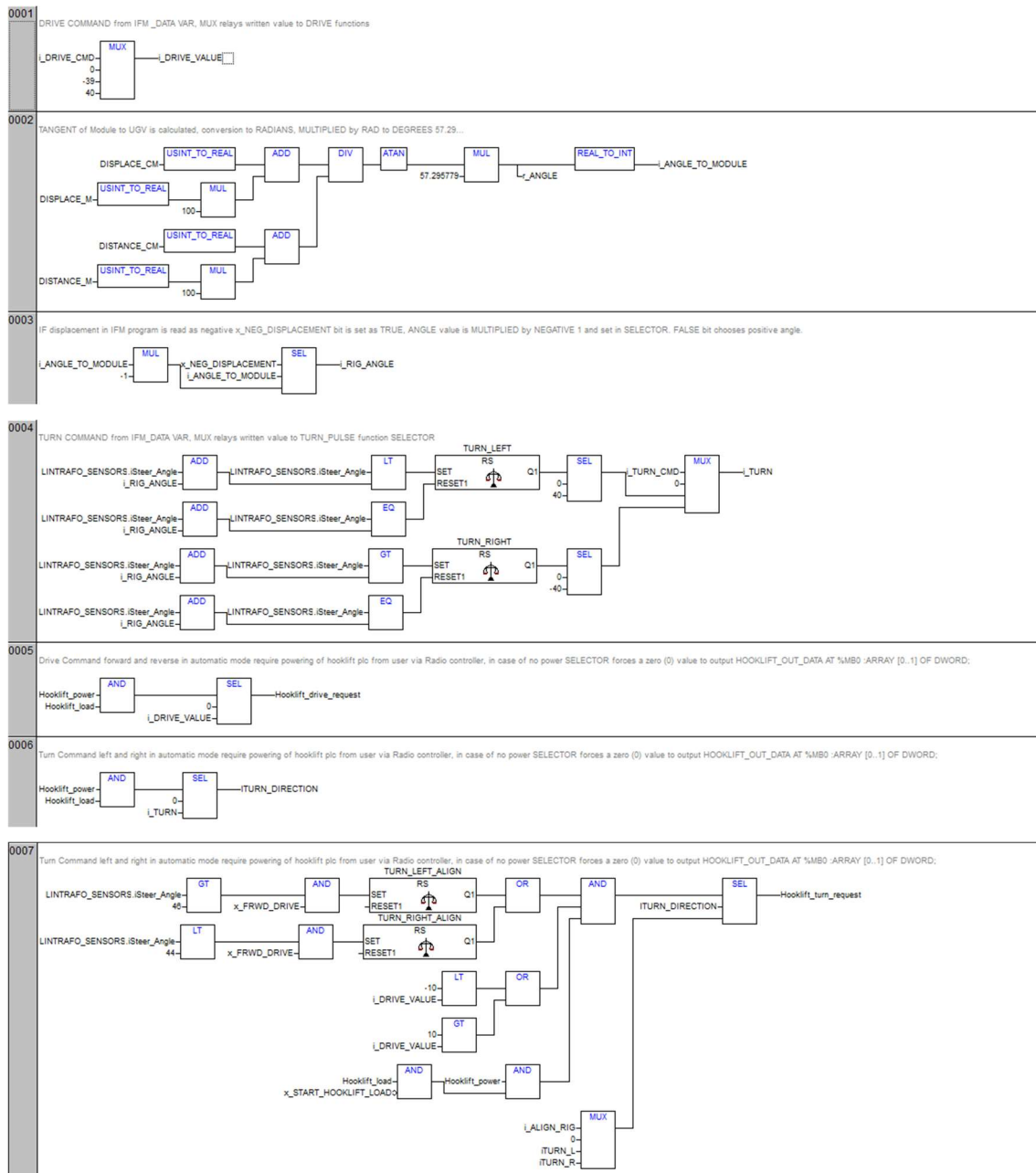
ST:llä luotu hydraulikkaventtiileiden ohjaus.


```

0001PROGRAM CONTROL_CONDITIONS
0002
0003(* IF Vehicle is within lifting range REVERSE DRIVE is not possible and iDRIVE is forced to ZERO*)
0004(* TRUE is relayed to HOOKLIFT OUT Tx Hooklift_raedy_to_load*)
0005IF AGV_DETECTED = TRUE THEN
0006    Module_Detected := TRUE;
0007
0008ELSIF AGV_DETECTED = FALSE THEN
0009    Module_Detected := FALSE;
0010
0011END_IF
0012
0013(* In the case of a detected module with reflectors and no requests for an unload function OR during the retrieval process, switching from reflector program to Normal Object program within the range of 135 to 150 cm*)
0014(*Detection of, in this case the front area of the Module AND without any load commands in process, the Drive request is REVERSE*)
0015
0016IF x_Hook_unload = FALSE AND
0017    (DRIVE_CMD = TRUE AND x_Hook_unload = FALSE) AND xSignal_UP = FALSE AND HOOK_LOAD.LOAD_FUNCTION.Q1 = FALSE THEN
0018    i_DRIVE_CMD := 1;
0019    x_FRWD_DRIVE := FALSE;
0020    (* ELISIF load function of tilting up is within range of 100 to 130 degrees, Drive request is FORWARD to align module with UGV*)
0021
0022ELSIF xSignal_UP = TRUE AND LINTRAFO_SENSORS.iAngle_Position >= 100 AND LINTRAFO_SENSORS.iAngle_Position <= 130 AND x_Hook_unload = FALSE THEN
0023    i_DRIVE_CMD := 2;
0024    x_FRWD_DRIVE := TRUE;
0025    (* In cases of no detection of reflectors, Objects too close observed in IFM code, an unload function in process or load function parameters claimed below, front end lifting, Drive request is NEUTRAL*)
0026    (*DRIVE_CMD is a command relayed over CAN from IFM AGV_programs, as of 26_2_2018 still testing if there is a need for an overlaying drive command*)
0027
0028ELSIF DRIVE_CMD = FALSE OR x_Hook_unload = TRUE OR (xSignal_UP = TRUE AND
0029    LINTRAFO_SENSORS.iAngle_Position >135 OR LINTRAFO_SENSORS.iAngle_Position <70) OR HOOK_LOAD.LOAD_FUNCTION.Q1 = TRUE OR
0030    LEVEL_CONTROL.FRONT_LIFT.Q1 = TRUE THEN
0031    x_FRWD_DRIVE := FALSE;
0032    i_DRIVE_CMD := 0;
0033
0034END_IF
0035
0036(* DRIVE_TURN_ALIGN requests are during the loading procedure when tilting is within range and aFORWARD DRIVE process is in action. Direction is based on the difference compared to rig angle sensor*)
0037(* In this case the sensor works within 0-90 degrees, 45 degrees being the desired value when front and rear UGV bodies are aligned*)
0038IF DRIVE_TURN.TURN_LEFT.ALIGN.Q1 = TRUE THEN
0039    i_ALIGN_RIG := 1;
0040
0041ELSIF DRIVE_TURN.TURN_RIGHT.ALIGN.Q1 = TRUE THEN
0042    i_ALIGN_RIG := 2;
0043
0044ELSIF DRIVE_TURN.TURN_LEFT.ALIGN.Q1 = FALSE AND DRIVE_TURN.TURN_RIGHT.ALIGN.Q1 = FALSE OR (DRIVE_CMD = FALSE AND x_Hook_unload = TRUE) THEN
0045    i_ALIGN_RIG := 0;
0046
0047END_IF
0048
0049(* DRIVE_TURN_DIRECTION requests are during the positioning procedure to a realized module. Direction is calculated in the DRIVE_TURN program depending on rig and modules relative position*)
0050IF DRIVE_TURN.TURN_LEFT.Q1 = TRUE AND x_Hook_load = TRUE THEN
0051    i_TURN_CMD := 1;
0052
0053ELSIF DRIVE_TURN.TURN_RIGHT.Q1 = TRUE AND x_Hook_load = TRUE THEN
0054    i_TURN_CMD := 2;
0055
0056ELSIF DRIVE_TURN.TURN_LEFT.Q1 = FALSE AND DRIVE_TURN.TURN_RIGHT.Q1 = FALSE OR x_Hook_unload = TRUE THEN
0057    i_TURN_CMD := 0;
0058
0059END_IF
0060
0061
0062(* IF the Object distance is within range and other requests are Zero, then UGV is raelised as being within a lifting range*)
0063IF i_TURN_CMD = 0 AND i_DRIVE_CMD = 0 AND DRIVE_CMD = FALSE AND DISTANCE_M = 0 AND DISPLACE_M = 0 THEN
0064    x_START_HOOKLIFT_LOAD := TRUE;
0065
0066ELSIF i_TURN_CMD = 1 OR i_TURN_CMD = 2 OR i_DRIVE_CMD = 1 OR DRIVE_CMD = 1 THEN
0067    x_START_HOOKLIFT_LOAD := FALSE;
0068
0069END_IF
0070
0071
0072
0073
0074
0075
0076
0077
0078
0079
0080
0081
0082
0083
0084
0085
0086
0087
0088
0089
0090
0091
0092
0093
0094
0095
0096
0097
0098
0099
0100
0101
0102
0103
0104
0105
0106
0107
0108
0109
0110
0111
0112
0113
0114
0115
0116
0117
0118
0119
0120
0121
0122
0123
0124
0125
0126
0127
0128
0129
0130
0131
0132
0133
0134
0135
0136
0137
0138
0139
0140
0141
0142
0143
0144
0145
0146
0147
0148
0149
0150
0151
0152
0153
0154
0155
0156
0157
0158
0159
0160
0161
0162
0163
0164
0165
0166
0167
0168
0169
0170
0171
0172
0173
0174
0175
0176
0177
0178
0179
0180
0181
0182
0183
0184
0185
0186
0187
0188
0189
0190
0191
0192
0193
0194
0195
0196
0197
0198
0199
0200
0201
0202
0203
0204
0205
0206
0207
0208
0209
0210
0211
0212
0213
0214
0215
0216
0217
0218
0219
0220
0221
0222
0223
0224
0225
0226
0227
0228
0229
0230
0231
0232
0233
0234
0235
0236
0237
0238
0239
0240
0241
0242
0243
0244
0245
0246
0247
0248
0249
0250
0251
0252
0253
0254
0255
0256
0257
0258
0259
0260
0261
0262
0263
0264
0265
0266
0267
0268
0269
0270
0271
0272
0273
0274
0275
0276
0277
0278
0279
0280
0281
0282
0283
0284
0285
0286
0287
0288
0289
0290
0291
0292
0293
0294
0295
0296
0297
0298
0299
0300
0301
0302
0303
0304
0305
0306
0307
0308
0309
0310
0311
0312
0313
0314
0315
0316
0317
0318
0319
0320
0321
0322
0323
0324
0325
0326
0327
0328
0329
0330
0331
0332
0333
0334
0335
0336
0337
0338
0339
0340
0341
0342
0343
0344
0345
0346
0347
0348
0349
0350
0351
0352
0353
0354
0355
0356
0357
0358
0359
0360
0361
0362
0363
0364
0365
0366
0367
0368
0369
0370
0371
0372
0373
0374
0375
0376
0377
0378
0379
0380
0381
0382
0383
0384
0385
0386
0387
0388
0389
0390
0391
0392
0393
0394
0395
0396
0397
0398
0399
0400
0401
0402
0403
0404
0405
0406
0407
0408
0409
0410
0411
0412
0413
0414
0415
0416
0417
0418
0419
0420
0421
0422
0423
0424
0425
0426
0427
0428
0429
0430
0431
0432
0433
0434
0435
0436
0437
0438
0439
0440
0441
0442
0443
0444
0445
0446
0447
0448
0449
0450
0451
0452
0453
0454
0455
0456
0457
0458
0459
0460
0461
0462
0463
0464
0465
0466
0467
0468
0469
0470
0471
0472
0473
0474
0475
0476
0477
0478
0479
0480
0481
0482
0483
0484
0485
0486
0487
0488
0489
0490
0491
0492
0493
0494
0495
0496
0497
0498
0499
0500
0501
0502
0503
0504
0505
0506
0507
0508
0509
0510
0511
0512
0513
0514
0515
0516
0517
0518
0519
0520
0521
0522
0523
0524
0525
0526
0527
0528
0529
0530
0531
0532
0533
0534
0535
0536
0537
0538
0539
0540
0541
0542
0543
0544
0545
0546
0547
0548
0549
0550
0551
0552
0553
0554
0555
0556
0557
0558
0559
0560
0561
0562
0563
0564
0565
0566
0567
0568
0569
0570
0571
0572
0573
0574
0575
0576
0577
0578
0579
0580
0581
0582
0583
0584
0585
0586
0587
0588
0589
0590
0591
0592
0593
0594
0595
0596
0597
0598
0599
0600
0601
0602
0603
0604
0605
0606
0607
0608
0609
0610
0611
0612
0613
0614
0615
0616
0617
0618
0619
0620
0621
0622
0623
0624
0625
0626
0627
0628
0629
0630
0631
0632
0633
0634
0635
0636
0637
0638
0639
0640
0641
0642
0643
0644
0645
0646
0647
0648
0649
0650
0651
0652
0653
0654
0655
0656
0657
0658
0659
0660
0661
0662
0663
0664
0665
0666
0667
0668
0669
0670
0671
0672
0673
0674
0675
0676
0677
0678
0679
0680
0681
0682
0683
0684
0685
0686
0687
0688
0689
0690
0691
0692
0693
0694
0695
0696
0697
0698
0699
0700
0701
0702
0703
0704
0705
0706
0707
0708
0709
0710
0711
0712
0713
0714
0715
0716
0717
0718
0719
0720
0721
0722
0723
0724
0725
0726
0727
0728
0729
0730
0731
0732
0733
0734
0735
0736
0737
0738
0739
0740
0741
0742
0743
0744
0745
0746
0747
0748
0749
0750
0751
0752
0753
0754
0755
0756
0757
0758
0759
0760
0761
0762
0763
0764
0765
0766
0767
0768
0769
0770
0771
0772
0773
0774
0775
0776
0777
0778
0779
0780
0781
0782
0783
0784
0785
0786
0787
0788
0789
0790
0791
0792
0793
0794
0795
0796
0797
0798
0799
0800
0801
0802
0803
0804
0805
0806
0807
0808
0809
0810
0811
0812
0813
0814
0815
0816
0817
0818
0819
0820
0821
0822
0823
0824
0825
0826
0827
0828
0829
0830
0831
0832
0833
0834
0835
0836
0837
0838
0839
0840
0841
0842
0843
0844
0845
0846
0847
0848
0849
0850
0851
0852
0853
0854
0855
0856
0857
0858
0859
0860
0861
0862
0863
0864
0865
0866
0867
0868
0869
0870
0871
0872
0873
0874
0875
0876
0877
0878
0879
0880
0881
0882
0883
0884
0885
0886
0887
0888
0889
0890
0891
0892
0893
0894
0895
0896
0897
0898
0899
0900
0901
0902
0903
0904
0905
0906
0907
0908
0909
0910
0911
0912
0913
0914
0915
0916
0917
0918
0919
0920
0921
0922
0923
0924
0925
0926
0927
0928
0929
0930
0931
0932
0933
0934
0935
0936
0937
0938
0939
0940
0941
0942
0943
0944
0945
0946
0947
0948
0949
0950
0951
0952
0953
0954
0955
0956
0957
0958
0959
0960
0961
0962
0963
0964
0965
0966
0967
0968
0969
0970
0971
0972
0973
0974
0975
0976
0977
0978
0979
0980
0981
0982
0983
0984
0985
0986
0987
0988
0989
0990
0991
0992
0993
0994
0995
0996
0997
0998
0999
1000

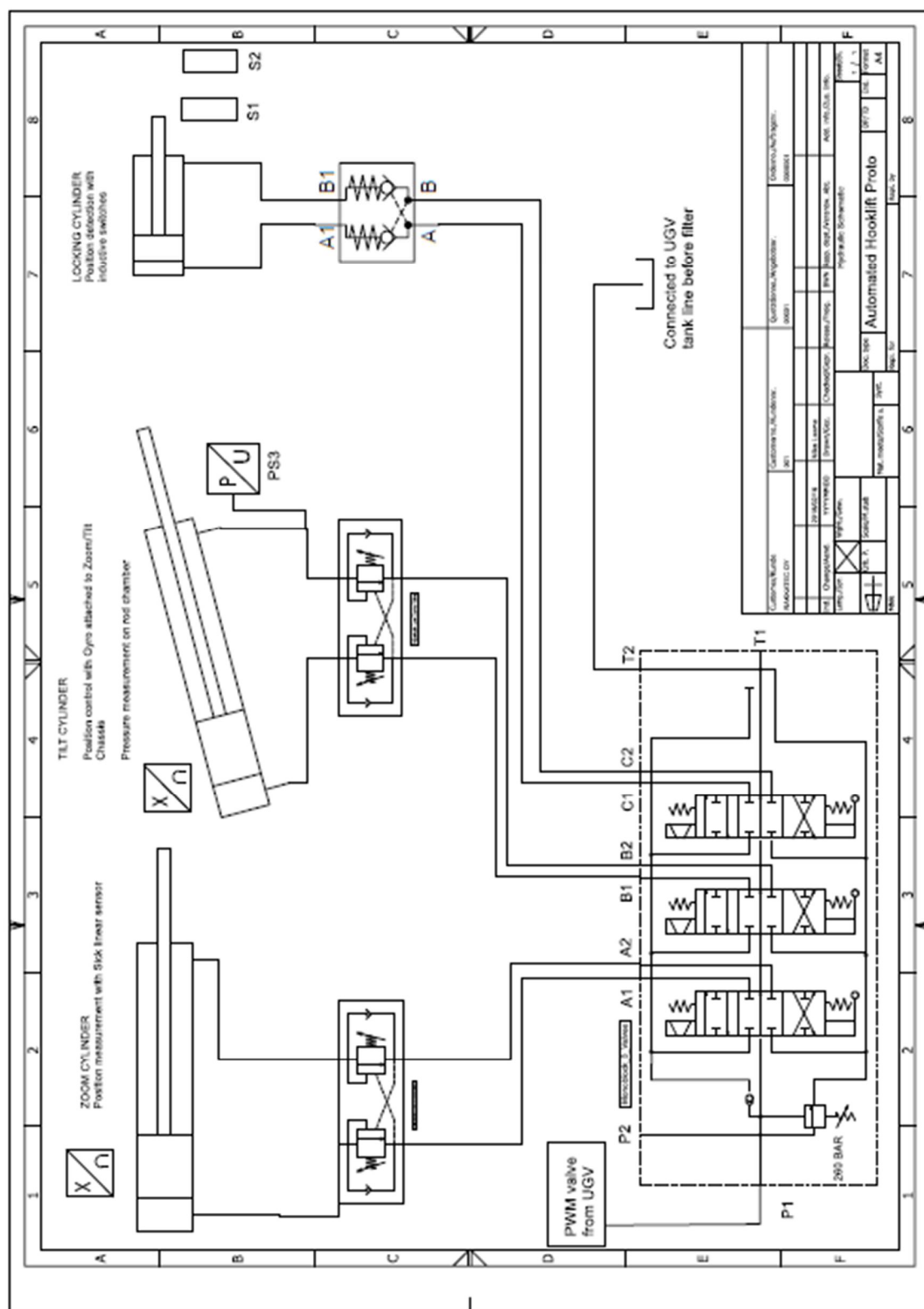
```

ST:llä luotu ajon, käännön ja lastauksen suoritus ehdot.



FBD:llä luotu alustan ja kohteen välisen aseman laskenta ja autonominen ohjaus.

Hydrauliikkakaavio



Sähkökytkentöjen kytkentälista

ELECTRICAL CONNECTION LIST									
Cable	position1	position 2	CAN_H	CAN_L	ground	signal	signal	REMARKS	battery + FUSE at Fusebox
MAIN POWER	FUSE BOX	UGV Main Battery Switch							
IFM CAN 1 + fuse 4	Power junction box	CR1083	grey n1	black n2	white			2,5 mm²	brown FUSE 4
TEC CAN_1	Power junction box	TEC	yellow n1	green n2					
TEC CAN_2	Power junction box	TEC	yellow n3	green n4					
Resistor 120 Ohm	Power junction box		n3	n4					
RS-232 RXD	Power junction box	TEC				red n5			
RS-232 TXD	Power junction box	TEC					green n6		
Ground-Frame	Power junction box	TEC			brown x2 n7			connected to UGV	
Battery 24V fuse 3	Power junction box	TEC							red x2 n8 FUSE 3
IFM supply 24V fuse 1	Power junction box	O3M Smart Sensor			blue	white			grey+brown n8 FUSE 1
IFM supply 24V fuse 2	Direct	Scanner Lighting			blue x2 n7		black		2X Brown FUSE 2
IFM GROUND	Direct	Scanner Lighting			2X BLUE				
outputs									
X2-1	PWM Valve UGV	TEC			IO ground-blue x 2	brown		Same as outputs	
X2-4	solenoid HY1	TEC			X2-13			Same as outputs	
X2-5	solenoid HY2	TEC			X2-13			Same as outputs	
X2-9	solenoid HY3	TEC			X2-28			Same as outputs	
X2-10	solenoid HY4	TEC			X2-28			Same as outputs	
X2-11	solenoid HY5	TEC			X2-29			Same as outputs	
X2-12	solenoid HY6	TEC			X2-29			Same as outputs	
X2-6	IO junction box	Inductive switches			blue			Same as outputs	
X2-8	IO junction box	Linear sensor			blue			Same as outputs	
X2-8	IO junction box	Inclination sensor			brown			Same as outputs	
X2-7	IO junction box	Pressure sensors			blue			Same as outputs	
X2-24	IO junction box	Hall-sensor			black			Same as outputs	
inputs									
X2-19 sig	IO junction box	Inductive OPEN			X2-30-blue	white		black NOT CONNECTED	
X2-14	IO junction box	Linear sensor			X2-31-brown	white		black NOT CONNECTED	
X2-15	IO junction box	Inclination sensor			X2-32-blue	black		white, NOT CONNECTED	
X2-18 sig	IO junction box	Inductive CLOSED			X2-30-blue	white		black NOT CONNECTED	
X2-20 sig	IO junction box	Inductive POSITION			X2-30-blue	white		black NOT CONNECTED	
X2-21 sig	IO junction box	Hall-sensor			IO ground-green	blue		white, NOT CONNECTED	
X2-25 sig	IO junction box	Pressure sensor PLUS			X2-30	white		black NOT CONNECTED	
X2-26	IO junction box	Pressure sensor MINUS			X2-30	option		option	